



N-1803

NCEL

November 1989

By R. L. Webster and P. A. Palo

Technical Note

Sponsored By Naval Facilities
Engineering Command

SEADYN90 USER'S MANUAL

AD-A217 229

DTIC
ELECTE
JAN 26 1990
S E D

ABSTRACT This manual describes the input structure to the latest release of the cable dynamics computer model SEADYN. SEADYN is a general purpose, finite-element FORTRAN code that models the static and dynamic responses of arbitrarily-configured cable structures, including effects due to: environmental loads, surface/bottom (with friction) interaction, variable length lines, moved boundaries, and strumming. Program features include: free-field input format; interfaces for user-specified drag coefficients, current fields, and time-variation functions; incremental and iterative numerical solution methods; RESTART capability; and user-defined output files. Example problems are included.

Mooring Line, Tension Structure, etc.

NAVAL CIVIL ENGINEERING LABORATORY PORT HUENEME CALIFORNIA 93043-5003

METRIC CONVERSION FACTORS

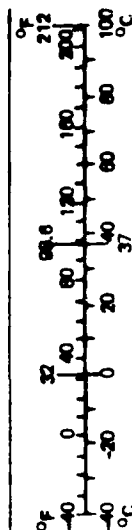
Approximate Conversions to Metric Measures

| Symbol | When You Know | Multiply by | To Find | Symbol |
|----------------------------|------------------------|----------------------------|---------------------|-----------------|
| LENGTH | | | | |
| in | inches | 2.5 | centimeters | cm |
| ft | feet | 30 | centimeters | cm |
| yd | yards | 0.9 | meters | m |
| mi | miles | 1.6 | kilometers | km |
| AREA | | | | |
| in ² | square inches | 6.5 | square centimeters | cm ² |
| ft ² | square feet | 0.09 | square meters | m ² |
| yd ² | square yards | 0.8 | square meters | m ² |
| mi ² | square miles | 2.6 | square kilometers | km ² |
| | acres | 0.4 | hectares | ha |
| MASS (weight) | | | | |
| oz | ounces | 28 | grams | g |
| lb | pounds | 0.46 | kilograms | kg |
| | short tons | 0.9 | tonnes | t |
| | (2,000 lb) | | | |
| VOLUME | | | | |
| tsp | teaspoons | 5 | milliliters | ml |
| fl oz | tablespoons | 15 | milliliters | ml |
| c | fluid ounces | 30 | milliliters | ml |
| pt | cups | 0.24 | liters | l |
| qt | pints | 0.47 | liters | l |
| gal | quarts | 0.96 | liters | l |
| ft ³ | gallons | 3.8 | liters | l |
| yd ³ | cubic feet | 0.03 | cubic meters | m ³ |
| | cubic yards | 0.76 | cubic meters | m ³ |
| TEMPERATURE (exact) | | | | |
| °F | Fahrenheit temperature | 5/9 (after subtracting 32) | Celsius temperature | °C |

* 1 in. = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 288, *Units of Weights and Measures*, Price \$2.25, SO Catalog No. C13.10-288.

Approximate Conversions from Metric Measures

| When You Know | Multiply by | To Find | Symbol |
|-----------------------------------|-------------------|------------------------|-----------------|
| LENGTH | | | |
| millimeters | 0.04 | inches | in |
| centimeters | 0.4 | inches | in |
| meters | 3.3 | feet | ft |
| meters | 1.1 | yards | yd |
| kilometers | 0.6 | miles | mi |
| AREA | | | |
| square centimeters | 0.16 | square inches | in ² |
| square meters | 1.2 | square yards | yd ² |
| square kilometers | 0.4 | square miles | mi ² |
| hectares (10,000 m ²) | 2.5 | acres | |
| MASS (weight) | | | |
| grams | 0.035 | ounces | oz |
| kilograms | 2.2 | pounds | lb |
| tonnes (1,000 kg) | 1.1 | short tons | |
| VOLUME | | | |
| milliliters | 0.03 | fluid ounces | fl oz |
| liters | 2.1 | pints | pt |
| liters | 1.06 | quarts | qt |
| liters | 0.26 | gallons | gal |
| cubic meters | 35 | cubic feet | ft ³ |
| cubic meters | 1.3 | cubic yards | yd ³ |
| TEMPERATURE (exact) | | | |
| Celsius temperature | 9/5 (then add 32) | Fahrenheit temperature | °F |



REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

| | | | | | |
|---|---|--|--------------------------------------|--|--|
| 1. AGENCY USE ONLY (Leave blank) | | 2. REPORT DATE November 1989 | | 3. REPORT TYPE AND DATES COVERED Final — Jun 88 to Jun 89 | |
| 4. TITLE AND SUBTITLE SEADYN90 USER'S MANUAL | | | | 5. FUNDING NUMBERS PR-44-085 WU — DN666284 | |
| 6. AUTHOR(S) R.L. Webster* and P.A. Palo | | | | | |
| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Civil Engineering Laboratory Port Hueneme, CA 93043-5003 | | | | 8. PERFORMING ORGANIZATION REPORT NUMBER TN-1803 | |
| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Naval Facilities Engineering Command Alexandria, VA 22332 | | | | 10. SPONSORING/MONITORING AGENCY REPORT NUMBER | |
| 11. SUPPLEMENTARY NOTES *Consulting Engineer Brigham City, Utah 84302 | | | | | |
| 12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited. | | | | 12b. DISTRIBUTION CODE | |
| 13. ABSTRACT (Maximum 200 words) This manual describes the input structure to the latest release of the cable dynamics computer model SEADYN. SEADYN is a general purpose, finite-element FORTRAN code that models the static and dynamic responses of arbitrarily-configured cable structures, including effects due to: environmental loads, surface/bottom (with friction) interaction, variable length lines, moved boundaries, and strumming. Program features include: free-field input format; interfaces for user-specified drag coefficients, current fields, and time-variation functions; incremental and iterative numerical solution methods; RESTART capability; and user-defined output files. Example problems are included. | | | | | |
| 14. SUBJECT TERMS Cable dynamics, cable statics, ocean structures, finite-element | | | | 15. NUMBER OF PAGES 208 | |
| | | | | 16. PRICE CODE | |
| 17. SECURITY CLASSIFICATION OF REPORT Unclassified | 18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified | 19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified | 20. LIMITATION OF ABSTRACT UL | | |

FOREWORD

The process of learning to use a complex computer program is most often frustrating. Regardless of the care taken to document and explain the various features of the program, a new user is easily confused by unfamiliar terminology and descriptions of the methods employed. Usually, the developers have become so conversant with the problems addressed by the program and the way that they have interpreted and solved them, that it is difficult to communicate with the uninitiated. They simply forget how far they have come from the beginning. Often the appropriate beginning point can be found and quick learning obtained through dialogue between the developer or an experienced user and the novice. Unfortunately, the written page does not provide for such feedback. One should keep in mind that it is at least as difficult to write a manual that adequately describes a program as it is to learn to use the program.

| | |
|----------------------|-------------------------------------|
| Accession For | |
| NTIS GRA&I | <input checked="" type="checkbox"/> |
| DTIC TAB | <input type="checkbox"/> |
| Unannounced | <input type="checkbox"/> |
| Justification | |
| By _____ | |
| Distribution/ | |
| Availability Codes | |
| Dist | Avail and/or Special |
| A-1 | |



PREFACE

The SEADYN cable dynamics computer program was developed over a 10-year period primarily by the Naval Civil Engineering Laboratory (NCEL) under the sponsorship of the Naval Facilities Engineering Command (NAVFAC). This document is a revision of TN-1775, July 1987, to reflect the latest release named SEADYN90. SEADYN is the largest of several cable analysis programs developed under the Large Displacement Cable Dynamics project. These programs have been validated by laboratory and at-sea experiments and allow for the confident analysis of arbitrarily configured cable structures subject to a wide variety of environmental and system loads.

This report is one in a series of three used to document the SEADYN computer program. This set of reports consists of:

- (1) SEADYN User's Manual: Describes the program input format and external file requirements, and briefly discusses use of the program. Example inputs are presented.
- (2) SEADYN Theoretical Models: Describes the finite-element formulation, implementation of particular submodels (strumming model, linearizations in the frequency domain solution, etc.), and numerical solution techniques (Ref 1).
- (3) SEADYN Programmer's Reference: Describes SEADYN87 coding structure, logic, memory usage, and required system routines (Ref 2).

References 3 and 4 provide the following information and data on SEADYN-related topics: Data on cables, chain, and other SEADYN input parameters (Ref 3) and a summary of the comparisons between SEADYN predictions and measured data (Ref 4).

This version of SEADYN has deleted the frequency domain analysis of moored vessels that has been in earlier versions. All references to that capability have been removed from the documentation. New features have been added and existing features have been enhanced or simplified in a number of instances (see Section 2.4). Users of earlier versions of SEADYN are advised to review this manual carefully to note some input format changes and generalizations.

THE U.S. GOVERNMENT ASSUMES NO LIABILITY
FOR ANY LOSS OR DAMAGE
RESULTING FROM THE USE OF THIS PROGRAM.

CONTENTS

| | Page |
|---|------|
| 1.0 INTRODUCTION | 1 |
| 2.0 CAPABILITY OVERVIEW | 2 |
| 2.1 General Features | 2 |
| 2.2 Structure Modeling Features | 2 |
| 2.3 Forcing Functions and Analysis Options | 3 |
| 2.4 New and Modified Features in This Version | 5 |
| 3.0 THEORY OVERVIEW AND MODELING SUGGESTIONS | 8 |
| 3.1 Modeling Lines and Cables | 8 |
| 3.2 Static Ship Model and Loads | 11 |
| 3.3 Solution Methods | 11 |
| 3.4 Structure Model | 14 |
| 4.0 INPUT FILE DATA STRUCTURE | 19 |
| 4.1 Model Formulation (Problem Data Block) | 19 |
| 4.2 Loads and Analysis Deck (SAO Data Block) | 21 |
| 4.3 RESTART Formulation | 22 |
| 5.0 FREE-FIELD INPUT RULES | 23 |
| 6.0 PROBLEM DEFINITION DATA | 26 |
| 6.1 Title Record Set | 28 |
| 6.2 PROBLEM | 29 |
| 6.2.1 BLOC | 30 |
| 6.2.2 BODY | 31 |
| 6.2.3 DRAG | 33 |
| 6.2.4 ELEMENT | 37 |
| 6.2.5 FLOW | 41 |
| 6.2.6 FLUID | 43 |
| 6.2.7 INVENTORY | 44 |
| 6.2.8 LIMIT | 45 |
| 6.2.9 LLOC | 47 |
| 6.2.10 MATERIAL | 48 |
| 6.2.11 NGEN | 51 |
| 6.2.12 NODE | 53 |
| 6.2.13 PAYOUT | 56 |
| 6.2.14 SHIP | 58 |
| 6.2.15 STRUM | 61 |
| 6.2.16 TENSION | 62 |
| 6.2.17 TFUNCTION | 63 |

CONTENTS (cont'd)

| | Page |
|---|------|
| 6.3 RESTART | 69 |
| 6.4 TABLE | 72 |
| 6.5 EDIT | 75 |
| 7.0 SUBANALYSIS OPTION (SAO) DATA | 76 |
| 7.1 DEAD, LIVE, TSSS, DYN Data Set | 77 |
| 7.1.1 CURRENT | 78 |
| 7.1.2 FIX | 80 |
| 7.1.3 FREE | 81 |
| 7.1.4 IMPACT | 82 |
| 7.1.5 INITIAL | 83 |
| 7.1.6 KEEP | 84 |
| 7.1.7 LOAD | 85 |
| 7.1.8 LVARY | 87 |
| 7.1.9 MOVE | 88 |
| 7.1.10 OUTPUT | 91 |
| 7.1.11 PAYOUT | 94 |
| 7.1.12 SAVE | 96 |
| 7.1.13 SBUOY | 97 |
| 7.1.14 SOLUTION | 98 |
| 7.1.15 STEP | 104 |
| 7.1.16 SURFACE | 105 |
| 7.1.17 TIME | 106 |
| 7.1.18 WIND | 107 |
| 7.2 MODE | 108 |
| 7.2.1 MSOL | 109 |
| 7.2.2 FIX | 110 |
| 7.2.3 FREE | 111 |
| 7.2.4 MOVE | 112 |
| 7.3 CHEK | 115 |
| 7.3.1 ANCH, BUOY, LINE | 116 |
| 7.4 END, NEW | 118 |
| 8.0 EXAMPLE PROBLEMS | 119 |
| 8.1 Towed Body Example | 119 |
| 8.2 Buoy Relaxation Example | 126 |
| 8.3 Anchor Last Example | 128 |
| 8.4 Payout From a Moving Ship Example | 130 |
| 8.5 Vessel Mooring Example | 132 |

CONTENTS (Cont'd)

| | Page |
|---|------|
| 9.0 SUMMARY OF MAJOR DIAGNOSITC MESSAGES | 139 |
| 9.1 System Description Checks | 140 |
| 9.2 Subanalysis Option Errors | 142 |
| 9.3 Solution Option Execution Messages | 144 |
| 10.0 REFERENCES | 150 |
| APPENDIXES | |
| A - Ship Wind and Current Loads Data File | A-1 |
| B - Description of External Plot Data File | B-1 |
| C - Default Drag Coefficients | C-1 |
| D - User-Supplied Subroutines | D-1 |
| E - Restart File Structure | E-1 |
| F - Ship's Wind and Current Loading Functions | F-1 |
| G - Built-In Load Functions For Ships | G-1 |
| H - Mooring Component Inventory | H-1 |
| I - Description of the External Motion Data File | I-1 |
| J - Program Size Limitations and Storage Requirements | J-1 |

1.0 INTRODUCTION

The SEADYN computer program simulates the responses of cable and truss type systems in an ocean environment. Such structures often appear deceptively simple. Cable structures, even a simple catenary span, are highly nonlinear and careful modeling is required to obtain a solution. Adding the offshore environment greatly compounds the problem. This computer program provides extensive capability for dealing with these problems and includes considerable flexibility and variety in the solution methods.

SEADYN is a finite element program that employs simple truss-type elements and catenary elements. These line elements can be used repetitively to describe long mooring lines, intricate truss structures, or complex cable systems. SEADYN includes lumped body models to simulate buoys, anchors, weights, etc. in the structure. These lumped masses are treated as point loads that represent weight/buoyancy and fluid drag effects. SEADYN also models rigid bodies that represent ships, barges, platforms, and mooring buoys. These are distinct from the previously mentioned lumped bodies since they provide for multiple line attachments and full six degree-of-freedom body response calculations. Two additional unique features of the program are the ability to represent variable length lines (payout/reel-in) and the component adequacy checks (tests buoys, anchors, and lines for potential underdesign).

The approach taken in this manual is to briefly outline the nature of the nonlinear problem, describe in simple terms the pertinent features of the nonlinear solution methods, present the input form for SEADYN, and illustrate the modeling process with some simple problems. Exposition of the governing equations and theoretical details is contained in Reference 1.

2.0 CAPABILITY OVERVIEW

The SEADYN program is designed primarily to analyze the static and dynamic responses of underwater cable and truss structures. The major features of the program are summarized below.

2.1 General Features

3-D Large Displacement Response of Cable and Truss-Type Structures Using the Finite Element Method. SEADYN is capable of simulating large nonlinear structural displacements as well as small linear displacements of cable structures.

Free Field Input. The program has a free-field input format. A keyword card begins each data set, such as NODE or ELEMENT. The data is entered after the keyword in free format.

Modular Format. The input data file is divided into two sections: the structure's model description and the analysis sequences. Only key nodes and elements are required for input. SEADYN will generate the intermediate nodes and elements and estimate line tensions. An equilibrium state for gravity loading will be determined through a DEAD analysis. Optional static and dynamic analyses may follow.

Restart Capability. The initial configuration can be saved, and a restart option can be used to apply different loads to the saved structure. Other analysis solutions can also be saved and used as reference in restarts. Economies in computer time and costs can be realized with this modular format for problem definitions.

2.2 Structure Modeling Features

The modeling options in SEADYN have been made as generic as possible so that a variety of cable structures can be simulated: instrument arrays, vessel moorings with one or many mooring lines, riser moorings, cable laying and pickup sequences, towed bodies, or cable structure deployment scenarios. A description of the structure modeling features and options follows.

Lumped and Rigid Bodies. Discrete bodies, such as buoys, sinkers, and instrument packages, can be modeled as either lumped bodies or rigid bodies. Lumped bodies provide only loads and mass to a node. Rigid bodies have a 6 degree-of-freedom response.

Truss and Catenary Line Elements. Cables are modeled using two types of elements: straight-line truss elements (also called simplex elements) or catenary-shaped elements. These elements ignore bending and torsional effects. Only axial loads (tension and compression) are considered. The two element types are interchangeable and universal in that they can be mixed and used anywhere in the model. Truss elements can have any orientation in a three-dimensional space; therefore, they are used to model the effects of three-dimensional, nonplanar, nonuniform,

distributed loads (i.e. a subsurface current profile that changes speed and direction with depth) and point loads. Catenary elements can also have any three-dimensional orientation but the element must lie in a plane due to the constraints of a classical catenary shape; these elements can model planar uniform distributed loads (not necessarily in the vertical plane) and point loads.

Nonlinear Materials. Two basic material models are provided to represent static load behavior. One uses a load-strain tabular format for specifying nonlinear behavior. The other is a two-parameter model that presumes curve-fitting of empirical data using the form $T = ae^b$. Material damping effects can be included through a one-parameter (Kelvin damping) or a two-parameter (Reid-NOAA) viscoelastic model. A proportional damping option is also provided.

Node and Element Generation. As in other finite element method programs, nodes and elements can be generated where an incremental pattern exists so that only beginning and ending nodes and elements of a homogeneous line segment need to be defined.

Conditionally Imposed Boundaries or Restraints. Boundaries, such as the seafloor or water surface, can be defined. Nodes can be permanently fixed to a boundary, conditionally attached to a boundary until a load condition is exceeded, or stopped at a boundary if sinking or rising.

Default, Resident, or User-Defined Functions. Commonly used drag coefficients, fluid flow fields (i.e. subsurface current profile), and time varying load functions are resident in the SEADYN program. If the default or resident functions are not suitable, the user can define the desired function. This procedure is described in the Appendices.

2.3 Forcing Functions and Analysis Options

SEADYN can simulate steady-state equilibrium conditions, compute mode shapes and natural frequencies, process nonlinear dynamic time domain solutions for cable structures without a vessel, and perform a component adequacy check for a given structural state.

Static Equilibrium and Steady-State Forces. The steady-state or static solutions find an equilibrium position for the cable structure for steady loads such as gravity, wind, surface currents, subsurface currents, and point loads.

Mode Shapes and Natural Frequencies. The mode shapes and natural frequencies are calculated for any given configuration. Because mode shapes and natural frequencies are based upon the structure's mass and stiffness terms, they should be calculated for a range of expected conditions.

Approximate Time Domain Dynamics. An approximate nonlinear dynamic analysis, the Time Sequenced Static Solution or TSSS, is available as an inexpensive option to a full time domain solution. In TSSS, motions

implied by the MOVE or PAYOUT options are applied to the structure in small steps which are determined by user specified time intervals. The acceleration and mass terms are ignored. The solutions use the static LIVE analysis procedure and outputs are printed for user-specified time intervals. The accuracy of this solution method deteriorates as a function of time. This analysis option is excellent for observing trends before committing to a full nonlinear time domain analysis. Rigid bodies representing ships, platforms, etc., may be included in the model because this is a static procedure.

Nonlinear Time Domain Dynamics. The nonlinear dynamic time domain analysis calculates the response of the cable structure to time-varying loads. The program prints out a set of solutions as a function of time which can be envisioned as series of snapshots. Rigid body dynamic equations are not provided in SEADYN. This precludes transient computations with ship, platform, etc., components in the model.

Mooring Component Adequacy Check. The component adequacy check compares the break strength or holding capacity of the mooring components against the calculated loads. The mooring component information is found in a library resident in SEADYN.

Strumming. The LIVE, TSSS, and DYN SAO operations can also approximate the effects of line strumming by allowing for the adjustment of the drag loads.

Analysis Options. Each analysis option in SEADYN is referred to as a subanalysis option (SAO). A list of the SAO flags and a brief description follows:

- DEAD - Nonlinear static analysis to apply gravity, buoyancy loads, and point loads.
- LIVE - Nonlinear static analysis to apply wind, surface and subsurface currents, gravity, buoyancy loads, and point loads.
- MODE - Calculates mode shapes and natural frequencies for a specified structure state. The MODE SAO can be used after any of the other SAO options listed before continuing on the next SAO option. This option provides information only.
- TSSS - Approximates nonlinear dynamics by neglecting acceleration and mass terms. Solves for configuration changes due to payout or reel-in of lines, to move boundaries, or imposed motion. Generates a LIVE solution for each time step requested.
- DYN - Nonlinear dynamic time domain analysis which solves for responses due to time-varying loads, currents, motions, line lengths, and lumped body impact.
- CHEK - Evaluates the adequacy of various mooring components in the currently defined state.

2.4 New and Modified Features in This Version

The following is a list of new features in SEADYN90.

1. Generation of an optional data file suitable for passing plotable information to a subsequent process. This optional plot uses the TABLE output templates to select the data. The OUTPUT record has been expanded to allow specification of the frequency of the output of the plot data. The format of the external plot file is described in Appendix B.

2. Nodes that are in contact with a bottom constraint are able to react to horizontal loads with a Coulomb friction force that allows sliding when the resistance capability is exceeded. The input for this option is given with the LIMIT data set.

3. The bottom constraint has been generalized to allow the bottom to be a tilted plane. Each LIMIT set can define an explicit plane that designated nodes will not be allowed to penetrate. If friction data is given for the LIMIT which is a tilted plane, any sliding will be along the plane until a lift-off condition is detected.

4. An option to allow the EDIT of some of the currently defined PROBLEM data between Subanalysis Option steps and after RESTARTs has been provided. This option is described in a new section under the EDIT keyword.

5. An output option has been added that reports the status changes for the surface and bottom nodal constraints. When selected, this option provides output whenever a node is released or has a constraint newly imposed. The minimum output reports the time and node number and indicates the release or hold status. When the node that has a bottom constraint imposed is also on a payout leg, additional output about the payout process is provided. (A node is determined to be on a payout leg when it is one of the ends of an element in the payout leg topology.) This printed information will be included in the standard output and optionally included on a special file that lists only the constraint changes. When selected, these status outputs will occur regardless of any other output options in effect during the solution.

6. Two new resident flow fields have been added. The first is a tabular function that allows arbitrary flow vector variations with depth. The second is a function with an exponential decay with depth. A new feature has been added to the CURR data record in the SAO data sets that allows a weighted combination of flow fields to be processed to get the flow vector.

7. Two new time function definition options have been added. The first allows time function data to be read from a table on a pre-generated data file. The second function allows access to motion RAOs that have been generated in a format defined by the SCORES ship motion program. This option then generates a table of time response data that is representative of the RAOs interacting with a user specified wave spectrum. Either of these options can be used as specified motion components for nodes.

The major modifications of options in SEADYN90 are:

1. The deletion of the FREQ option. Its utility and validity had not been established over ten years and it was unlikely to happen in the fore-seeable future. It was decided to archive that portion of the coding, stop maintaining that capability, and simplify the support of the program.
2. The restrictions on node numbering relative to slave and master nodes has been removed.
3. The requirement for assigning two consecutive nodes to represent six degree of freedom nodes has been removed. In order to do this it has been necessary to change the format of the NODE, INIT, LOAD, and MOVE input data. Although not seen by the user, this change has led to a major change of the internal data structure of SEADYN.
4. The DRAG loading models for line elements have been generalized to allow forms other than those based on the independence principle. New options have been added to the built-in functions and a new user-defined option has been added.
5. Upper and lower bounds have been added to the MODE option so that output of the mode shape and frequency data will only be done for modes between those user selected limits.
6. The MOVE data record has been modified to accomodate nodal movement data given by the new time function tables. It also supports a new capability to describe node point motion in terms of tow point velocity and heading data.
7. The default specification of the initial reference length for the PAYOUT topology has been reversed. The default value for MITNOT (a value of 0 in Word 8 of the PAYOUT data record in the PROBLEM data set) means that mitosis will not occur on the initial active element until its length has grown to exceed the initial element length plus the mitosis length. This assumption is also imposed when the mitosis length is altered by the PAYOUT record in any DYN or TSSS data set. Normal mitosis (division when the element length exceeds twice the mitosis length) occurs on subsequent elements. Assigning MITNOT a nonzero value will produce a normal mitosis on the first active element.
8. The standard output for the status of payout ends has been expanded to include data about the accumulated length of the payout leg (as implied by the payout topology data) and the velocity at the payout point.
9. All input and output related to reference configuration updates has been deleted. SEADYN now does configuration updates at the end of each solution step, i.e. it employs the incremental moving reference form of Lagrangian kinematics. The so called total Lagrangian approach is no longer supported.

Table 2-1 provides a summary of the important features of SEADYN.

Table 2-1. SEADYN Capability Summary

General

- 3D large displacement response of cable and truss-type structures using the finite element method
- Can include lumped bodies, six degree-of-freedom rigid bodies (ships, platforms, etc., for statics only) and fluid-solid interactions
- Staged format, sequential analysis for statics and dynamics
- Treats nonlinear materials with internal damping, nonconservative loads, and nonlinear constraints

Special Features

- Variable length lines to simulate payout/reel-in dynamics
- Automatic estimation of drag coefficient amplification to approximate strumming
- Restart options
- Component adequacy checks using design rules
- Plotting interface
- Free-field input format
- Catenary element for treating bottom interaction

Load/Boundary Conditions

- Gravity/buoyancy loads in water and air
- Arbitrary point loads and flow fields
- Wind and surface current loads on rigid bodies
- Built-in or user-supplied drag functions with flow/response-dependent amplification
- Arbitrary time variations (built-in or user-supplied)
- Moved boundaries
- Conditional constraints for surface and bottom limits
- Bottom limits may be imposed with or without friction on a horizontal or sloped bottom

Static Solution Methods

- Residual feedback method (incremental self-correcting)
- Modified Newton-Raphson (various forms)
- Viscous relaxation method

Dynamic Solution Methods

- Nonlinear Transient
 - Newmark's β (residual feedback form)
 - Direct Integration Method (a multi-parameter predictor/corrector)
 - Time sequenced static solutions (quasi-static)
- Frequency Domain
 - Mode shapes and frequencies

3.0 THEORY OVERVIEW AND MODELING SUGGESTIONS

The approach taken in SEADYN to model cable and mooring systems is a discrete element method and the lumped parameter method. Lines are modeled with finite elements and bodies are lumped at node points.

The only deformable components in the system are the cable elements. Any component which cannot be modeled as a line element is assumed to be a nondeformable body. They are lumped at a node and may have a point effect or act as a rigid connector for attached lines. No dynamic equations are included for the rigid connectors, so their use is limited to static analyses.

3.1 Modeling Lines and Cables

Element Types and Properties. The SEADYN program represents a general spatial arrangement of cable and truss components as a collection of simple elements. Only one material type is allowed for each element. SEADYN has two element forms: a truss element which is a straight line between two nodes and a catenary element which has a catenary shape between two nodes.

The truss element is straight before and after deformation of the structure. The element uses the instantaneous distance between nodes and the unloaded length to determine strain. The truss element can model the effects of three-dimensional, nonplanar, nonuniform, distributed loads (such as a subsurface current that varies speed and direction with depth) and point loads. A linear variation of flow velocity and direction along the element is assumed.

The catenary element uses classical catenary equations to determine its shape between two nodes. Each element lies in a plane oriented in a three-dimensional space. Its shape and stiffness are dependent upon the forces at the nodes. Line stretch is included and, if applicable, the amount of line lying on the seafloor is determined. The catenary element can only model point loads and planar, uniform, distributed loads. Only approximate mass relations are provided for treating the dynamics of line pick-up and lay-down, so this element should be used with caution in transient dynamic solutions.

Determining the Number of Elements per Line. A line of elements has negligible bending resistance, i.e. no moment is supported at the nodes. Straight line segments with constant tensions are low order approximations of a flexible catenary line which has constantly varying slopes and tensions along its length for most cases. Regions where large curvature, highly variable tensions, or boundary interaction (such as line laying on the seafloor) are expected should be approximated with more elements than those regions where tensions vary slightly and lines are nearly straight. Distributed loads, such as subsurface currents, also require more elements. Parametric studies have indicated that a minimum of 5 to 8 elements are needed to accurately model lines with no distributed loads; a minimum of 10 to 12 elements is needed to accurately model lines that are subjected to distributed loads. These figures are only intended as a rule-of-thumb estimate. The selection of the number of elements should be based on several trials of element size and distribution. When a finer mesh gives essentially the same tensions and displacements, then the correct number of elements has been found which will control discretization errors.

Fluids. The cable system may be totally immersed, suspended between two fluids (i.e. air and water), or fluid effects may be ignored. Fluid effects are assumed to be uncoupled from the structure. Such things as flow alteration from structure movement or flow blockage are not dealt with. The effects of the structure moving from one fluid into another are not treated. Once an element has been assigned to a fluid, it is assumed to be in that fluid regardless of its position.

Bottom Boundary Interaction. Catenary elements are recommended for bottom interaction where lines are layed down or picked up from the sea-floor boundary because the exact length of line required can be determined. Very long catenary elements will introduce some error in transient solutions due to the condensedness of the mass lumping assumption. Truss elements must be either on or off the seafloor; partial lengths of the element cannot be laid on the boundary. Truss elements interacting with the bottom may include the influence of friction. The friction forces are lumped at the nodes on the bottom. A simple Coulomb friction model is used. In addition to the friction effects, it is possible to assign an alternate drag function for fluid load computations on truss elements that are on the bottom. The bottom may be represented as a horizontal plane or simple tilted plane. Bottom friction effects are not simulated for the catenary elements. If a truss element is used and element discretization is not fine enough, oscillating behavior may occur which leads to solution divergence.

Lumped and Rigid Bodies. Lumped bodies are assumed to have insignificant spatial dimensions so that their effect on the kinematics of the line are negligible. They are simply lumped at nodes where they produce mass and drag load effects. Nodes have three degrees-of-freedom, one for each displacement direction. Rigid bodies (mooring buoys, ships, platforms, etc.) are interfaced to cable elements using multiple attachment points. Rigid bodies require six degrees-of-freedom, three displacements and three angular components. Other nodes on the rigid body are slave nodes. Nodal slave/master constraints are imposed to imply rigid body kinematics. Slave nodes are constrained to move as though rigid links exist between them and the master pair of nodes. These slave/master constraints are operational for both static and dynamic solutions, but there are no dynamic equations for the rigid bodies. Unless custom interfaces are made to SEADYN that provide those equations, transient solutions with rigid bodies are not valid.

Distributed and Point Loads. Very abrupt load or displacement changes should be introduced incrementally using the IVARY record. Be careful not to introduce a load that will cause a physical instability, such as a load that is misaligned and causes an unintended rotation.

A common mistake in using SEADYN occurs when loads or constraints imposed in the preceding SAO data set are suddenly omitted in the next analysis. The load and displacement data defined at the beginning of each SAO analysis must be consistent with the total values obtained at the end of the preceding SAO analysis.

Applying loads which pull a buoy underwater leads to a sudden change in buoyancy loads if the surface current and subsurface currents are not well matched. For example, a buoy on the surface only has loads from a surface current and wind. If the surface current is large enough to push

the buoy under (due to the restraint of the mooring line), an equivalent subsurface current is required to keep the buoy submerged. If the subsurface current is smaller, then the buoy will oscillate between the surface position to being just submerged. The problem can also be resolved by removing surface loading from the buoy.

Users should be careful to put the proper variation codes on the loads for each SAO option. The MNR solution has an autostep selection process which works only when load variability is indicated. The force residual is tested when a new step is begun. If the force residual norm is greater than 3.0 then the size of the step is scaled down by the ratio of 3.0 and the residual norm. This helps to get an initially stable step size for the loads. The run will abort when it finds no change in the residual after the auto-step scaling process. In this case the KEEP option may not be appropriate because it preserves the variation codes.

Materials. The cable material is assumed to be nonlinear and elastic (no hysteresis).

Two methods are used to relate line tension to line strain. The first form is tabular. A series of loads and strains are entered which will be interpreted as linear segments approximating the load-strain curve. The second is a two-parameter curve fit.

The material constitutive relations are used in two situations in SEADYN. The primary use is to compute element internal loads (tensions) for the strains implied by node positions. A reverse process is used on initial configuration input. If an equilibrium configuration is input (either known or guessed), the unstrained lengths and the strain are calculated for the given nodal positions. If the unstretched element lengths and nodal positions are given, the strains and tangent modulus are calculated.

The computation of the unstretched and stretched lengths can be a source of error if the lines are very stiff (high EA) or the loads are small. This is because the small difference in strain cannot be detected due to the limits of the computer's precision.

Geometric Stiffness and Nonlinearity. The concepts of geometric stiffness and nonlinearity are illustrated below. A catenary changes its geometric configuration as well as load distribution to accommodate applied displacements or loads.

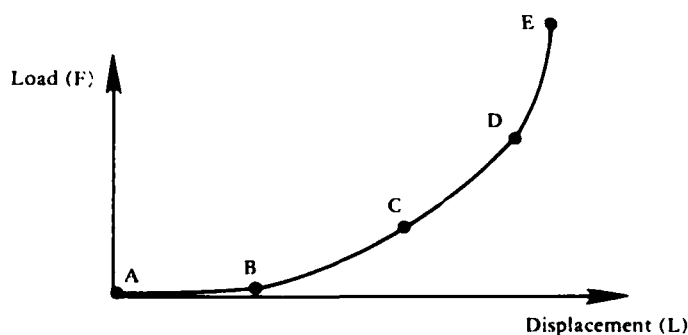


Figure 3-1: Load-deflection curve for a catenary.

A load-deflection curve for a catenary line is shown in Figure 3-1. The catenary line is essentially slack for the states between A and B. The top of the catenary line can be moved a great distance with very little change in the line tensions. The catenary line begins to develop some stiffness in the region BC. As the top of the line is displaced, the line tension also increases but not in a linear fashion. The position of the line also changes significantly. As the top of the line is displaced further, the system develops more stiffness as can be evidenced by the rapidly increasing tension in region CD relative to region BC. In region CD, the catenary geometry is beginning to approach a straight line. Between D and E, the line is straight and any further displacement results in axial strains and reorienting the line.

3.2 Static Ship Model and Loads

Vessel Geometric Description. If a vessel mooring is being modeled, the vessel is described using the SHIP record. The vessel's length, beam, draft, and displacement are required input. If wind loads are to be calculated for the vessel, wind areas must be input. SEADYN calculates the current areas from the length, beam, and draft. Since the static load functions deal only with surge, sway, and yaw, it may be necessary to restrain the other three degrees of freedom. Linearized ship restoring coefficients can be input as desired. Otherwise the ship will be fixed in heave during static analyses. Roll and pitch constraints must be imposed by user input if they are desired.

Wind and Current Loads. SEADYN requires the user to enter a file which describes the vessel response to wind and surface currents at various headings. This information is entered in fixed format anywhere in the PROBLEM data block after the PROBLEM data set or it may be extracted from a previously saved file. The format is given in Appendix A. See Reference 5 for suggestions on how to calculate these loads.

Axis/Coordinate System. The global axes used in the problem definition are defined in the PROBLEM data set. The axis that coincides with gravity is specified in the input and the right hand rule is assumed to define the other two axes. The vessel position and rotation relative to these global axes is defined in the NODE data set. This input establishes the relationship between the user selected global axes and the specific rigid body local coordinate system using Euler angles.

3.3 Solution Methods

General Solution Techniques. The majority of numerical solution techniques used in SEADYN can be classified as initial value methods. This means that a solution step proceeds from a state where all pertinent data are presumed to be known to a state where estimates are made of the effects of loading changes using some sort of predictor. When this estimate is within certain reasonable bounds of accuracy it can be improved by iterative corrections. In those situations where the initial state is not sufficiently described or where it represents an unstable state, the predictor is usually very inaccurate, if not undefined. SEADYN provides various means of dealing with this problem, but unfortunately there is no powerful method that works every time.

The various solution methods are described in the following paragraphs. The discussion focuses on difficulties that can be encountered in highly nonlinear or poorly posed problems and possible adjustments.

Static Solution Methods

Residual Feedback Method (RFB). This is an incremental self-correcting procedure that presumes loads or imposed displacement are applied in a sequence of increments. The first increment is based on the initial linearized stiffness of the system. Each additional step adjusts the specified load increment by an estimate of the equilibrium error from the previous step. The stiffness matrix is recalculated at each step and reflects the nonlinearities of the last step. This method is the least expensive, the most numerically stable, and the least accurate. A stable configuration is required to start from, since the method is very sensitive to the equilibrium of initial system state. The RFB method requires the user to specify the number of increments to be used to apply the loads. Only that number of increments specified is performed with no iterations. This method can fail if the stiffness matrix becomes singular or ill-conditioned. The final state can have large equilibrium error since a full iteration to convergence is never done. More accurate solutions can be obtained by increasing the number of steps. Its advantage is that it is quite stable and can span large static motions.

Modified Newton Raphson Method (MNR). This method can be used in a fully iterative (single-step) form or an incremental-iterative form. The method iteratively evaluates the difference between the external forces applied at the nodes and the internal reactions to search for the displaced state that satisfies equilibrium. The tangential stiffness matrix is used to estimate the displacement changes for each iteration. This matrix can be recalculated at each step or at user-specified intervals. The MNR method is conditionally stable. It can diverge if the predictor and convergence accelerators are inefficient or inappropriate. Various schemes are used to increase the rate of convergence when the step sizes are too small or numerical damping is too large. Convergence is determined when the displacement and residual force balance between external and internal forces are small. The user can specify the convergence tolerance criteria, the frequency of tangent matrix stiffness calculations, the number of iterations, and the character of extrapolation and convergence accelerators. Default values are taken if no specification is made for these parameters. This method offers opportunity for tuning the solution for dealing with special behavior, but it requires a level of sophistication and understanding of nonlinear procedures well beyond the casual user.

Viscous Relaxation Method (VRR). This is the most robust (and expensive) solution method. It is a generalized form of the Newton-Raphson method, which can automatically adapt the characteristics of the solution steps to the behavior being sensed. (An artificial time parameter is used to produce load steps or iterations.) On each load step, the VRR method iterates while adjusting the damping level and pseudo-time step

to move the equilibrium state. As convergence is achieved, the method degenerates to the Newton-Raphson method with a stiffness matrix evaluation at each iteration. It is often capable of getting solutions when others fail. The VRR method fails when there are excessive number of iterations or inappropriate selection of control parameters. The user can select the initial damping level, step size, number of load increments, and convergence tolerances. Default values are provided, but these parameters are problem dependent. Convergence is signalled by a very low value of nodal velocities or displacement changes and a low force residual between internal and external forces. It is the default (used when no other is selected) solution method for all static analyses.

NOTE: All of these static solution methods treat nonlinear behavior, but none can guarantee unconditional convergence. If a convergent solution is obtained, there is no guarantee that it is the only convergent solution. The analyst must review the results for plausibility on all nonlinear solutions.

Dynamic Solution Methods

Time Domain Analyses. Time domain analyses are fully nonlinear. Two basic solution methods are available for numerical integration of the nonlinear time domain equations. Both solutions are based on a generalized form of the Newmark difference equations (Reference 8). At present there are no time domain rigid body equations available in SEADYN so that nonlinear time domain solutions for ships, platforms, and/or buoys are not valid without some custom modifications to the code.

Implicit Residual Feedback Method (RFB). This is an implicit integration scheme that follows the more traditional Newmark format of solving a set of simultaneous algebraic equations at each time step. Payout/reel-in and moving boundary options have not been implemented in this solution method. Specifications of three integration parameters and the time step size is required. The method is strongly stable but can be inaccurate for large time steps.

Direct Integration Method (DIM). This is a predictor-corrector technique that does not require the formation of a stiffness matrix. Specification of three integration parameters and iteration controls is required. Time step size can be specified or calculated by the solution routine. The iterative corrector is conditionally convergent and requires strict upper bounds on the time step. This is the default transient method and its use over the RFB method is recommended.

Mode Shapes and Natural Frequencies. This analysis uses the Jacobi method to give information about the cable system's natural frequencies and mode shapes. It can be invoked after any other analysis option. Recall that the natural frequencies and mode shapes are dependent upon the cable system's mass and stiffness. The system's natural frequencies and mode shapes will change for slightly different structural configurations or load conditions. This information is not used elsewhere by SEADYN. Only a diagonal mass matrix is used and no correction is made for the lack of tangential added mass on the cable elements.

All other dynamic options make this adjustment. Mode analyses do not require load specifications, but the displacement constraints in the preceding SAO option must remain in force. If these displacements were imposed with a MOVE record, then a FIX record or a MOVE record will be required in the MODE data set. If rigid bodies are included in the model, all six degrees of freedom should be fixed, because mass data for rigid bodies are not available.

3.4 Structure Model

Node Numbering and Minimizing Bandwidth. The solution of simultaneous equations in SEADYN follows a Gaussian elimination procedure that attempts to minimize computer storage by taking advantage of equation symmetry and bandedness. This means that the amount of storage required and the matrix operations are sensitive to the way the nodes are numbered. As a general rule, the largest difference between node numbers defining an element determines the bandwidth. The smaller the bandwidth, the smaller the computer CPU costs. In calculating the maximum node number difference, the master node number (with 6 degrees of freedom) must be used in place of the slave nodes. Another general rule for node numbering is to number the softer (less stiff) components first. This reduces numerical error propagation in the analysis. It should be noted that all solution methods in SEADYN, except the DIM method, are bandwidth sensitive.

Errors that can occur because of node numbering can show up as a singular system of equations. The message printed would be "SOLUTION FAILED DUE TO A ZERO PIVOT" (see Section 9.3). This message also results from input errors and unstable structures, one should not be hasty in concluding it is a result of accumulated numerical error. A specific situation where this can occur is in the static analysis of a multileg mooring of a ship. The ship's response may be stiffer than the lower portions of the mooring legs. If the ship's nodes are numbered before the mooring legs, numerical error propagation can cause a singular matrix error at the lower ends of the lines. Numbering the nodes so that the ship is last will remove the problem.

Relative Stiffnesses Between Structure Components. In modeling cable structures, moorings in particular, one often encounters portions of the structure with stiffnesses much lower than the rest of the system (i.e., heavy chain vs. synthetic lines). Lines carrying very low tension have very little stiffness. When the line is at a point on its load-deflection curve where large displacements result in small tension increases, the line stiffness (given by the load-displacement slope) is small. Additional loads that produce tension will stabilize the structure. The additional loads will move the line farther up on its load-displacement curve where displacement causes a proportional increase in line tensions and the line stiffness is larger.

Angular motions of rigid bodies, such as mooring buoys or ships, can also cause instabilities. The restoring moment to an angular motion is provided by a line tension at its attachment point. If the system has low line tensions, convergence may be troublesome as there is little tension to provide a restoring moment. See Vessel Angular Responses section that follows. More detailed discussion can be found in Reference 22.

The problem of low stiffness components is compounded by numerical errors associated with solving the stiffness equations. A mixture of soft and stiff elements leads to numerical ill-conditioning, which can be compounded by the sequence of equation processing. This sequence is determined by the order of the node numbering. Ideally, the soft components should be numbered before the stiff ones. Unfortunately, one cannot afford the luxury of optimum ordering because it can greatly increase the bandwidth, increasing the solution costs. Consult Reference 10 for more on equation ordering and solution errors.

Boundary Conditions. The structure model should be constrained between natural boundaries imposed by the water surface and bottom boundaries. One does not expect buoys to rise out of the water or anchors to sink beyond the seafloor (within reasonable amounts). The SEADYN program assumes that boundaries are planar. The surface is assumed to be a horizontal plane. The bottom may be horizontal or a set of simple tilted planes. Checks are made at each step to see if nodes are within the imposed limits. To reduce the number of calculations, the user specifies which nodes shall be checked against which conditions. Nodes may not be assigned to more than one limit description. When a specified node is within a limit tolerance, it is constrained. For buoys at the water surface boundary, the node is held fixed in the vertical direction but is free for lateral movement. All three components of motion may be fixed for anchors, friction conditions can be imposed or only the normal component of motion can be held. Whenever the line tensions exceed the weight or buoyancy of a body, the constraint is released (i.e., a sinker is picked up from the bottom). Line nodes on the bottom with no anchor or sinker should be constrained only in the vertical direction or have friction limits imposed.

Boundary conditions are defined using the LIMIT record. Boundaries are defined as buoyant limits or as weight limits. The LLOC record is used to specify which nodes and bodies are not to exceed the limits specified.

A very common error leading to a singular stiffness matrix is an incomplete set of boundary conditions. Boundary conditions must be given to restrict rigid body motions of the entire structure and establish structural stability. If bodies or nodes are not marked as boundary limited and exceed the limit, anomalous behavior can occur.

Vessel Angular Responses. This version of SEADYN allows only static solutions with rigid bodies such as buoys, platforms, ships, etc. SEADYN treats yaw, roll, and pitch equally using full Euler angle kinematics. There is an upper bound of 10 degrees for angle changes during any solution iteration. Whenever a solution step estimates an angle change larger than this limit, the response is scaled back to the limit. The angle limit is imposed because the displacement increments are computed from a local linearization process. Any large angle change resulting from a linearized process would be inaccurate. When the limit is too large, the angle changes can lead to slack conditions on the upper elements and/or erratic behavior of the surface/bottom limits. With the angle limit kept sufficiently small it is not necessary to use any angle damping. In particularly sensitive situations it may be necessary to make the angle limit quite small and increase the iteration limit to

accommodate small changes in each iteration. When the angle limit is too small the solution progress is slow and stiffness matrix updates occur at every step regardless of the value given in the SOLUTION record Word 15 because angle scaling occurs on every iteration.

The temporary fixing of an angular response due to an alternating moment is a very effective feature. The MNR Solution will do this automatically when angle responses are unstable. It can lead to dramatic improvements in the convergence rate. It can also be confusing when it does not converge. If the limit number of iterations is exceeded while an angle is fixed, the residual force and displacement norm values may be below the convergence tolerance. In addition to the norms being sufficiently small, all held angles must be released before convergence is recognized. This is because all held components have their contribution to the residual set to zero.

Describing an Initial Geometric Shape. The easiest way to describe an initial geometric shape is to give the spatial coordinates of key nodes and the horizontal tension of the elements. With this information, SEADYN can generate the intermediate nodes and elements and determine the element lengths.

Most often, the user does not have an equilibrium state from which to start. Unstretched element lengths are known and the general form of the structural layout can be defined, but the nodal positions and tensions are unknown.

The user can guess nodal positions and the unstretched element lengths with estimated tensions and look at the output to see if the element lengths are too long or if tension is too high. The input can be adjusted accordingly and the problem rerun. This will be an iterative process but satisfactory configuration can usually be determined in a few trials.

The analyst should keep in mind that the finite element model is an approximation of the actual structure. Even if an exact description is available, it would not represent a convergence equilibrium state. Some minor adjustment in node positions and element tensions will inevitably be calculated.

Solution Stability of Large Cable Structure Displacements. Since SEADYN deals with large deflection effects, the position and velocity of all nodes and the unstretched lengths of all elements must be considered in each step of the solution. This is a result of geometric nonlinearity, and it poses some problems not encountered in small displacement analyses. In static analyses, where inertia effects are ignored, it is possible to have a set of unstretched lengths and nodal positions that represent an unstable configuration as an initial model. Unless the specified loads and solution procedures produce appropriate movements to modify the position of the nodes and reorient the elements, the structure is not capable of providing a static load path between the points where loads are applied and where they are supported. Another form of instability is the more familiar buckling instability in which an apparently stable structural configuration will suddenly deform to a radically different shape with only minor changes in loading.

Buckling or Slack Lines. Even when a stable structural state has been obtained, it is possible to develop solution instabilities while subsequently applying additional loads and/or movement. These problems result from physical instabilities in the structure, such as buckling, or from an inconsistent set of pretensions that cause a portion of the structure to go slack, causing numerical instabilities such as a singular matrix. Often both situation occur together. In general, classical bifurcation buckling behavior does not occur with cable systems. Incremental/iterative solutions would show a buckling instability as a large change in deflection for a small change in load. The load-deflection plot would have a strong curvature with the slope tending to zero. Numerically, this means the stiffness matrix has some terms which are very small relative to the other terms and is ill-conditioned (nearly singular).

Slow Convergence of Solution. If numerical damping (specified on the SOLUTION record) is too large or the angle limit is too small for the particular problem, movement will be sluggish and the force residual (difference between internal and external forces) will not change significantly in any one iteration. This can occur in the VRR solution if the structure is stiff because of high loads or material EA's. The VRR solution will signal this situation by repeated output of the SLOW CONVERGENCE message with no INCREASING NORM messages. Very slow convergence of the VRR method suggests that the algorithm wants near-zero damping, which is the Newton-Raphson method. The appropriate action is to repeat the analysis with a smaller damping or a large angle limit.

Alternatively, the MNR method could be tried with default parameters. The next choice would be MNR with a stiffness matrix calculation of each iteration (the default is every 5 iterations). Sluggish behavior of the MNR method also means numerical damping is too high or the angle limit is too low.

The more common situation is divergent behavior evidenced by very large, and often increasing, oscillating results in the iterations. This usually means that the finite element model does not have enough elements. The solution is somewhere between two calculated positions and the model will keep oscillating between them.

Also, a poor starting point that requires significant angular movement can cause oscillating behavior. In this case, large damping may help. The VRR approach will generally work much better than the MNR approach in these cases since it can more readily adapt to the problem by internal numerical damping and step-size adjustments.

The adaptive features of the VRR method have been developed to recognize a particular divergent pattern and then construct a strategy to react to it. It is possible that in other situations the action taken will be inappropriate or that a divergent pattern is undetected. These situations usually produce copious messages with a mixture of INCREASING NORM and SLOW CONVERGENCE, and no clear pattern of the reduction of the force residual is obtained. The solution will then fail by running to the limit number of iterations, time limits, or output lines limit (see SOLUTION record). If this occurs, the pattern of messages and the values of the residual force norms and velocity norms should be studied before attempting a rerun. Sluggish behavior will be indicated by small velocity and large residual force norms with small position changes. If the pattern

shows large velocity norms and no definite residuals force norm pattern, an increase in damping could be effective. Applying the load in increments with the STEP record is also effective.

If very erratic norm changes occur and the values are very large, one should review the problem input data to see if anything is wrong. If not, a different strategy should be considered. This may be a change in the initial state input and/or a different mix of analysis parameters. The possibility that the load level requested produces a physical instability should be investigated. Very abrupt load or displacement changes should be introduced incrementally.

Time Domain Analysis. The most effective solution for the time domain dynamics is the DIM record. Generally the default parameters will be appropriate. The choice of a time step is the main concern. The DIM method will select its own time step if none is given. One problem that occurs for strongly nonlinear dynamics is that the appropriate time step may vary with time. A new upper limit on the time step is calculated only when there is a signal that the analysis is not converging. The signals are:

- a. A displacement change norm exceeding 1.E12
- b. A large number of iterations without convergence
- c. A persistently increasing displacement change norm

In these situations, the time step's upper bound will be recalculated and the step size reduced. It is possible that the problem will not be detected early enough to avoid divergence. Time domain analyses which make repeated time step changes should be rerun with the step forced to remain below the range of changes. This is done by specifying DT on the TIME record and setting JMPDT to -1 on the SOLUTION record.

Time domain analyses without material damping tend to be sensitive to step size and will produce spurious oscillations if the step size and/or convergence tolerance are not appropriate. Even with damping, erroneous oscillations can be induced by setting convergence tolerance (DERR) too high. The correct value of DERR is somewhat problem dependent. The default value of 0.001 is an average value. It should not be set higher without some verification that the results are acceptable. It may be necessary to set it lower. One example of this is when the time step is desired to be much smaller than the upper bound, and strong nonlinearities are present (i.e. payout or bottom interaction). There is a tradeoff between step size, DERR, and the number of iterations to convergence. Although the optimum mix is hard to determine, experience has shown that DT and DERR values that give three to eight iterations per step are reasonable for strongly nonlinear problems.

4.0 INPUT FILE DATA STRUCTURE

The SEADYN program uses a free-field input format. Information describing the program and subanalysis options (SAO) is organized into modular blocks of data sets. General rules for entering the data records are given in Section 5.0.

For discussion purposes, the following terminology will be used:

| | |
|----------------|---|
| data deck: | the entire input file, all the data records |
| problem block: | the data records that describe the cable structure model |
| SAO block: | the data records that describe the loads and analysis methods to be used on the cable structure |
| data set: | the data records used to describe a subset of information, such as the NODE record and all the nodal information that follows |
| line: | a single line of data that may be up to 80 characters |
| data word: | the value to be assigned to an input variable |
| data record: | a logical collection of data words that describe an input entity, e.g. the input for a single node. |

The data deck always begins with at least one title line but there is no limit to the number of title lines allowed; the end of the title record is signified on the last title line by the ";" or "\$" character. The "*" character can be used to put in comments on a record after the data or to insert comment lines. A NEW record signals a return to problem definition and an arbitrary number of title lines can again be used. Either the PROBLEM or RESTART data set must follow the title record. The key words used to enter the data are limited to 10 characters but only the first four are required for input, such as PROB or REST. The basic program flow is represented in Figure 4-1.

The free-form input allows the data set for a ship load file to be read in the required fixed-field format. This data can be included anywhere in the model description after the PROBLEM data set. The fixed format is described in Appendix A.

4.1 Model Formulation (Problem Data Block)

The required sequence for the problem block is shown below. Only a few records are required. The rest are optional and are included only if applicable. Specific examples are given in Section 8.0 with example data decks for typical problems.

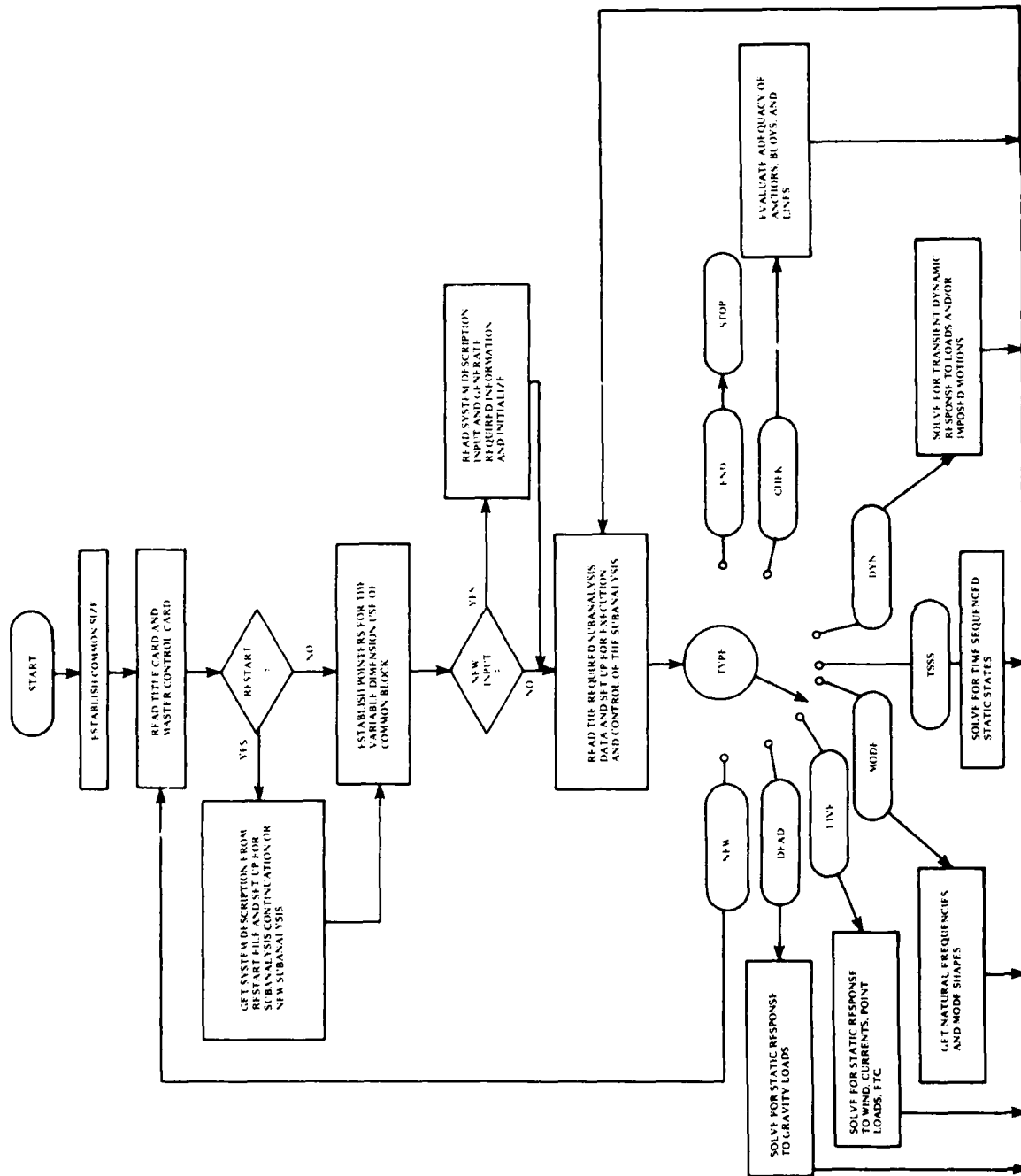


Figure 4-1. Macro flow chart of SEADYN.

Required RecordsOptional Records

| | |
|----------|---------------|
| Title \$ | FLUI |
| PROB | DRAG |
| NODE | BODY and BLOC |
| ELEM | LIMI and LLOC |
| MATE | NGEN |
| | FLOW |
| | INVE |
| | STRU |
| | SHIP |
| | TENS |
| | PAYO |
| | TABL |

(No record is required to signal the end of the problem data block, proceed to the SAO data block)

4.2 Loads And Analysis Deck (SAO Data Block)

The sequencing requirements for the SAO analyses follow a logical progression of loading. Unless one is absolutely certain that a configuration has been entered in perfect equilibrium, the DEAD analysis should always be first. This allows the program to adjust line lengths, tensions and nodal positions, and determine a stable equilibrium condition to which the desired loads can be applied. The exception to this is the case where no gravity loads are desired. Word (3) on the PROB record is set to zero, and the DEAD solution is omitted.

After a DEAD solution, any of the other options can be selected. Some common progressions are:

DEAD, LIVE, DYN
DEAD, DYN
DEAD, LIVE, MODE
DEAD, LIVE, TSSS
DEAD, LIVE, CHEK
DEAD, LIVE, DYN, NEW, REST, EDIT, DYN

Below is a listing of the load and solution type options for each SAO type.

DEAD - FIX, FREE, LOAD, LVARY, KEEP, MOVE, OUTPUT, SAVE, SOLUTION, STEP

LIVE - CURRENT, FIX, FREE, LOAD, LVARY, SURFACE, WIND, KEEP, MOVE, OUTPUT, SAVE, SOLUTION, STEP

TSSS - CURRENT, FIX, FREE, LOAD, LVARY, MOVE, PAYOUT, SURFACE, TIME, WIND, KEEP, OUTPUT, SAVE, SOLUTION

DYN - CURRENT, FIX, FREE, IMPACT, INITIAL, LOAD, LVARY, MOVE, PAYOUT, SURFACE, TIME, WIND, KEEP, OUTPUT, SAVE, SOLUTION

MODE - FIX, FREE, MOVE, OUTPUT, MSOLUTION

END - end of SAO data block

NEW - used in place of end; next record is title record for new
PROBLEM or RESTART data block

4.3 RESTART Formulation

Title \$

RESTART

DEAD, 1... or LIVE, 2... or DYN, 3... or NEW, 4...

SAO Data Set' (if needed)

END

5.0 FREE-FIELD INPUT RULES

The following special characters are recognized by the free-field input routine:

- \$ Record Terminator Flag
Signals no more data to be read for the record being processed. Multiple records can appear on a single line separated by record terminators. Double-record terminators signal the end of a line, and any data following this are treated as a comment. Comments will be listed as part of the record but will not be transmitted to the data file.
- ;
Alternate Record Terminator Flag
Performs same function as \$
- COLUMN80 Default Record Terminator
Unless a prior termination or continuation is signaled, the end of line (COLUMN80) is taken as a record termination.
- , Word Delimiter (Separator)
Separates sequences of data entries in a record. Repeated delimiters produce zeros in the words. Zeros for needed input will cause default values to be used by the program. An initial comma produces a zero in the first word of the record. All words not explicitly defined are assumed to be 0.0. A comma can be used to signal multiple lines (continuation) in a single record. In this case only blanks can occur between the last comma and the end of the line being continued.
- BLANK Separator/Delimiter
Leading blanks are ignored. Once the beginning of a word is detected, a blank will terminate the word. Any blanks following a delimiter are treated as leading blanks for the next word. The following are equivalent:
- xx yy
xx , yy
xx,yy
- / Continuation Flag
Signals a word termination with the next word to be read from the next line. See "," for alternate continuation.
- W Word Position Flag
Used to override the word sequencing and shift to a new word in the record. Input then follows in sequence from the new word location. The new word number is given immediately following the W and before the next "," or blank.

For example: 1, 2, 3, W7, 1.

The W can be used as a delimiter of the previous word on all but the first word in a record. The combination ",W" is the same as W alone. The first word of a record is not checked for the W flag so ",W" must be used to skip to a new sequence from the first word position. Any W after the first word and before the record terminator will be interpreted as a position flag.

* Comment Record Flag

This character anywhere on a record will terminate it and the remainder of the line is treated as a comment. A "*" in column 1 produces no record, and is only a comment.

(Fixed Format Initiator Flag

This in column 1 of any record after the title record signals that the records up to the next ")" record are in fixed format. These records are written on a special data file in BCD format.

) Fixed Format Terminator Flag

This in column 1 signals the end of a sequence of rigid format records.

Any record with a "\$", ";", or "*" in column 1 will be treated as a comment record. It will be listed but will not produce a data record.

Each free-field input deck is presumed to begin with one or more title lines. Title lines are read and listed until a specific record terminator is detected (\$ or ; but not COLUMN80). The line on which the terminator is detected will be used as a page heading for the run.

The apostrophe is a special character recognized by the free-field reader, but not needed in SEADYN input. Do not use the apostrophe anywhere except in comments and title lines.

The free-field reader processes the entire input deck and translates it into a series of data records. As noted above, a data record can span more than one line or there can be one or more records on a line. After the title record, the data records are assumed to be arranged in blocks headed by a flag record. Each flag record has a key word that is limited to ten alpha-numeric characters. Only the first four characters of the key word have meaning. For example, ELEMENTS could be shortened to ELEM to produce the same result. In those instances where the key word has only three characters, there must be a blank or word terminator after the third character. The specific data order applicable to the flag record is assumed until the next flag record is detected. Flag records must have the key word in the first word position.

Data records are assumed to be in floating point form unless a character is detected that is inconsistent for a floating point number. In this case, the word will be treated as a Hollerith word. All floating point words assume a decimal at the end of the word if none is given. Words actually intended to be integers are converted to a fixed-point form at the time they are used by the program. The actual length of a

data record is determined by the program using the free-field subroutine. The maximum length of a data record allowed by the free-field reader is 100 words.

The beginning and end delimiters for fixed format records must appear in column 1 of a distinct data record (individual line). This specifically requires that the previous record must have been appropriately terminated (no continuation). The () delimiters are the only things read on that record, and the next data processed are assumed to be on the next record. Only one rigid format data set is allowed in any run. A rigid format data set cannot be input before the initial title record set is completed.

6.0 PROBLEM DEFINITION DATA

There are two ways of providing problem definition data: with a PROBLEM or a RESTART data set.

The PROBLEM data set provides a complete specification of all of the nodes, elements, bodies, etc., that describe the problem at hand. The order of the data records is:

```
Title Record Set  
  
PROBLEM  
  
Problem data sets  
  
SAO Flag  
  
SAO data sets as needed  
  
END or NEW
```

The order of the problem data sets is not rigidly specified. Use the data sets as needed and follow logical sequences. For example: NGEN requires start and end node data that must be predefined by NODE and/or other NGEN data.

RESTART presumes a previously generated restart file is available that contains all of the problem description data and the results of the SAO calculations. RESTART can be used to continue the previous SAO with some of the options and parameters changed, if desired. At the completion of this SAO, any other appropriate SAO can be requested. Alternatively, RESTART can be used to establish a starting state for a new SAO. The appropriate form for RESTART input is:

```
Title Record Set  
RESTART  
Restart data set  
SAO flag  
SAO data sets as needed  
END or NEW
```

The data set descriptions are listed alphabetically by the flag record. A summary of flag records (four characters) follows:

Title

| | |
|------|--------------------------------------|
| PROB | New problem definition |
| BLOC | Body locations |
| BODY | Define lumped body table |
| DRAG | Define drag data sets |
| ELEM | Line or cable element definitions |
| FLOW | Flow-field library definitions |
| FLUI | Fluid media definitions |
| INVE | Modify component inventory |
| LIMI | Limit set definitions |
| LLOC | Limit locations |
| MATE | Material table definitions |
| NGEN | Generate nodes along a line |
| NODE | Node point definitions |
| PAYO | Set up payout topology |
| REST | Restart data |
| SHIP | Ship data definitions |
| STRU | Strum string definitions |
| TABL | Define optional output table entries |
| TENS | Initial tension input |
| TFUN | Time function library definitions |

The next sections present detailed instructions for inputting this information to SEADYN, including expected units where applicable (F = force, L = length, T = time).

6.1 Title Record Set

Any number of title lines can be used to begin the data set. At least one is required. The last title line must be terminated with a record terminator (; or \$). These record terminator characters cannot appear anywhere else in the title set.

The NEW flag record signals a new title record is to follow. Any number of title lines can be placed following a NEW record. The last title line must be terminated (as usual).

6.2 PROBLEM - New problem definition. Must immediately follow title records when it is used.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | NN | Number of nodes in model |
| 2 | NE | Number of line elements in model |
| 3 | IDIR | Gravity direction code (see Note 1) 0 - no gravity loading ± 1 - for X direction ± 2 - for Y direction ± 3 - for Z direction |
| 4 | IPRO | Input print option flag 0 - echo all input data with interpretation -1 - suppress input echo 1 - echo input plus print initial state data for nodes and elements and approximate transit times for each element |
| 5 | GRAV | Gravitational acceleration (L/T^2) Defaults to 32.174 ft/sec ² if IDIR \neq 0 and GRAV not given |
| 6 | NSFILE | Ship load data file flag (see Note 2) 0 - no data file N - ship load data for N ships are provided as rigid format input in this run -N - a data file with N sets of ship load data has been provided as TAPE 10. No ship load rigid format data is provided in this run. |

NOTES

1. It is assumed that the global axes are selected such that one of them coincides with the direction of gravity. Thus, IDIR = -2 means that gravity acts in the negative Y direction (i.e., +Y is up).

2. If a ship mooring is being modeled, the ship's static load data file (for wind and current loads) is expected on tape 10 or as part of the input problem block. Word(6)<0 means that the load data file was previously created and saved on tape 10. Word(6)>0 means that the load data file is in the problem block with rigid format. Appendix A describes the rigid format.

Every time a set of ship load data is in the problem block with rigid format, it is saved on tape 10. Future runs can use the saved file by referring to -N in NSFILE (where N is the number of ship load data sets on the file). If certain ships are to be frequently referenced, their load data could be saved on tape 10 and then recalled for all future runs. Thus, a library of ship loads can be provided in tape 10.

Only one rigid format data file is allowed in a problem block.

6.2.1 BLOC - Lumped/rigid body location specifications.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | LBODN (I) | Body table number (see BODY table) |
| 2 | LBEG | Beginning node number for body location(s) |
| 3 | LEND | Ending node number for body location(s) |
| 4 | LINC | Node number increment |

NOTES:

1. Cylindrical buoys use the attached elements to define the orientation of the long axis. The first two connecting elements are used. The slopes of these two elements will be averaged to get the orientation of the cylinder. If only one element is connected to the body, its direction will be used.
2. Mooring buoys (rigid bodies) require a minimum of three nodes in their definition. The buoy location must be a master node to give position and angle (six degrees-of-freedom, see NODE Note 1). Two additional slave nodes must be tied to the buoy. These two nodes are automatically defined from the first two nodes (in numerical order) slaved to the buoy location node. Recall that slave nodes need not have lines attached to them. The location node for a rigid body is the first of the master node (the one that gives the position coordinates). When less than three lines are attached to a mooring buoy, the roll motion about the line between the first two slave nodes is assumed to be fixed to avoid equation singularity problems.
3. Input for Words (3) and (4) is not required if only one body is being located by the record.
4. No more than 10 lines can be connected to any body location with limits imposed on it with an ILOC or BLOC data set.
5. The present version of SEADYN does not have a complete model for rigid bodies (using slave and master nodes) in the time domain solutions (DYN). The slave/master logic is operational in the time domain, but the dynamic equations for rigid bodies are incomplete.

6.2.2 BODY - Defines lumped body table. Use BLOC record to identify location of the bodies.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | I | Body table number |
| 2 | IDRB (I) | Drag function number (see Note 4) 0 - the default drag function will be used (see Appendix C) ≠0 - the given DRAG function number from the DRAG data set will be used. |
| 3 | BDW (I) | Body weight (weight is +; buoyancy is -) (F) |
| 4 | DBU (I) | Body diameter (L) |
| 5 | BLEN (I) | Body length (L) 0 - sphere or lump >0 - in-line cylindrical body |
| 6 | BAMC (I) | Added mass coefficient (Default for sphere = 0.5, for cylinder = 1.0) |
| 7 | BWND (I) | Wind drag coefficient for surface buoy (L^2) ($C_D * A_s$) |
| 8 | BSCD (I) | Surface current drag coefficient (L^2) ($C_D * A_s$) |
| 9 | BMOM (I) | Mass moment of inertia (FLT^2) |
| 10 | MEDMB (I) | Fluid medium in which ADM(I) is defined. See FLUID record (Default = 1). |

NOTES:

1. Lumped bodies can be spherical or an in-line cylinder. These bodies provide only loads/mass to the nodes where they are located. They do not have stiffness terms, nor do they have wave-induced loads in the FREQ subanalysis.

2. Bodies used as mooring buoys cannot be cylindrical. These bodies must be defined as "spherical" and have nodes assigned as described in Note 2 on the BLOC record.

3. Environmental loads on the body are determined from the relative velocity between the fluid and node, the dimensions and form of the body, and the drag coefficients (default or user defined).

4. Drag functions are defined by the DRAG record. A FLUI record must also be given for the input to be utilized. In the event that a FLUI record is given but no DRAG record is given, then the drag function number, Word (2), refers directly to the drag function code. In that case $IDRB(I) > 0$ refers to resident functions and $IDRB(I) < 0$ refers to user defined functions, given in the USRDRG subroutine. See Appendix D for the definition of USRDRG. This alternate form is provided to retain compatibility with earlier versions of SEADYN. The preferred method is to use the DRAG data record.

5. If the body is a buoy which can be on the surface or pulled under, care must be taken in defining the combination of surface and subsurface loads. (See the SURF, WIND, CURR and DRAG data records). Inconsistent definitions can lead to abrupt changes in loading which may cause solution instabilities. These instabilities most often result in the buoy oscillating between being on the surface and being submerged.

6.2.3 DRAG - Defines a drag function reference table.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | I | Drag function number (20 max) (see Note 1) |
| 2 | IDRGFN(I) | Drag function code (see Note 2) >0 resident function =0 default function <0 user-defined function |
| 3 | DRGPAR(1,I) | Optional drag function parameters (20 max) |
| . | | |
| . | | |
| . | | |
| . | | |
| 22 | DRGPAR(20,I) | (see Notes 2 and 5) |
| 23 | NODSLP(I) | Nodal slope option flag (see Note 7) (Default = 0 for do not invoke) (Nonzero means use element length weighted slopes at nodes) |

NOTES:

1. DRAG data is meaningful only if a FLUI record is provided. DRAG defines a library of drag functions to be referenced by MATE and BODY data. When no DRAG data set is provided to define a reference table, the drag function numbers used in the MATE and BODY data are interpreted as drag function code references.

2. The drag function code selects one of three types of drag functions for lines and bodies: resident, user-defined, or default subroutines. A positive function code refers to a resident functions (USRDRG subroutines which have been permanently incorporated into SEADYN), while a negative function code refers to a user-defined subroutine (USRDRG subroutines written, compiled, and loaded into SEADYN by the user). A request for a resident function beyond that currently defined will cause an abort of the run. The function parameters are used as appropriate for the resident functions, and they are provided in the calling sequence to USRDRG (see Note 5).

3. Default drag functions are provided in SEADYN. These are used when fluid data is defined and no other drag functions are selected. Two situations will result in the use of these default functions:

- a. No drag function number is provided in the MATE data for line elements or in the BODY data for lumped bodies.
- b. The value given for IDRGFN is zero.

The default drag coefficients are described in Appendix C.

4. The USRDRG user-defined subroutine format is described in Appendix D.
5. The resident drag functions at present are:

IDRGFN

- 1 (No DRGPAR parameters required)

For Spherical Buoys-

$$C_N = 0.47$$

For Cylindrical Buoys and Lines

$$C_N = 1.5 + 4/\sqrt{Re}$$

$$C_T = 0.02 C_N$$

For $Re < 0.10$

$$C_N = 13.$$

$$C_T = 10.$$

- 2 Constant Drag Function

DRGPAR(1,I) gives normal drag coefficient, C_N

DRGPAR(2,I) gives tangential drag coefficient, C_T

(These values are used regardless of the magnitude of the velocity components.)

See Note 6 for description of 100 series options that do not assume the independence principle.

6. Drag function codes (IDRGFN(I)) with a value of 100 or greater refer to function forms that do not assume the independence principle. Two possibilities for local coordinate systems for the drag functions are considered. The default form is a Tangential/Normal/Side form. The alternate is Drag/Lift/Side (i.e., horizontal/up/side). The two systems coincide for a zero element slope. Negative function codes with absolute values greater than 99 mean the USRLIN subroutine will be called. Positive values refer to one of the following resident functions: (F_i refers to the load function for the i direction; e.g., F_T means force in the tangential direction).

100 Fourier Coefficients for FT and FN, FS=0 based on the form in Reference 21.

101 Fourier Coefficient Form

Each load component has the form: $F_i = A_0 + A_1 \cos \phi + B_1 \sin \phi + A_2 \cos 2\phi + B_2 \sin 2\phi$ where ϕ is the element inclination angle in degrees

102 Polynomial Form

Each load component has the form: $F_i = A_0 + A_1\phi + A_2\phi^2 + A_3\phi^3 + A_4\phi^4$ where ϕ is element inclination angle in degrees

For all "100 series" options:

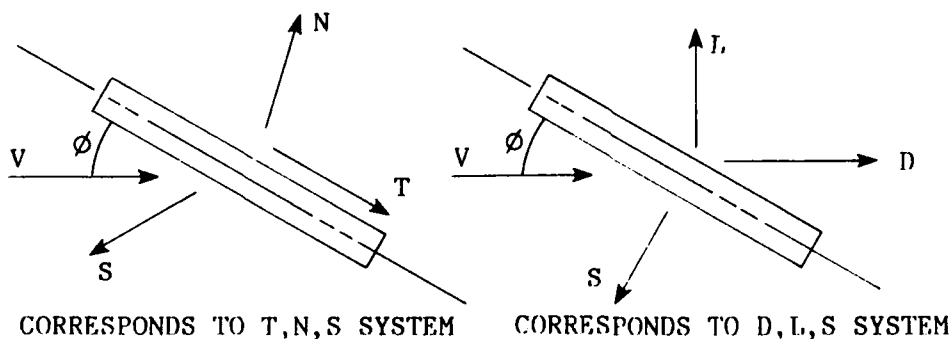
DRGPAR(1,J) Alternate Characteristics Length for Reynold's No.
A positive value gives the explicit value of the length.
A negative value gives the multiplier of the drag diameter from the MATE data. (default = drag diameter from MATE data will be used)

DRGPAR (2,I) DRAG function number for computing NORMAL drag coefficient (default $C_N = 1.0$ except for 100, which uses DTNSRDC f_n .)

(Option 100 uses no more of the data fields)

DRGPAR(3,I) DRAG function number for computing SIDE coefficient (default $C_S = 0.0$)

DRGPAR(4,I) Local Coordinate System Code,
0 means T,N,S: Tangent,Normal,Side
1 means D,L,S: Drag,Lift,Side



DRGPAR(5,I) up to DRGPAR(20,I) give the optional input.

for 101 5- 9 give A_0, A_1, B_1, A_2, B_2 for F_T or F_D
 10-14 give A_0, A_1, B_1, A_2, B_2 for F_N or F_L
 15-19 give A_0, A_1, B_1, A_2, B_2 for F_S

for 102 5- 9 give $\Lambda_o, \Lambda_1, \Lambda_2, \Lambda_3, \Lambda_4$ for F_T or F_D
 10-14 give $\Lambda_o, \Lambda_1, \Lambda_2, \Lambda_3, \Lambda_4$ for F_N or F_L
 15-19 give $\Lambda_o, \Lambda_1, \Lambda_2, \Lambda_3, \Lambda_4$ for F_S

7. Line elements normally assume that the line slope is constant and defined by a straight line between the two nodes at the ends of the element. If Word 23 is nonzero, the line slope will be replaced by two approximate slopes that represent the slopes at each end node. The approximation is obtained from a weighted average of the straight line slopes from the two elements that connect to the node. The weighting is done according to the reciprocal of the element lengths. If a long and a short element connect to the node, the slope for the shortest element will dominate the average. If more or less than two elements connect at the node, the straight line slope for the current element will be used at that node.

6.2.4 ELEMENT - Line or cable element definitions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | I | Element number |
| 2 | IT (1, I) | Node number for first end of element |
| 3 | IT (2, I) | Node number for second end of element |
| 4 | --- | Not used at present but the word position must be accounted for. |
| 5 | IT (4, I) | Material number |
| 6 | IT (5, I) | Element code (see Notes 2 through 7 for catenary elements) <ul style="list-style-type: none"> 0 - truss or simplex element which cannot support compression 1 - truss or simplex element which can support compression -1 - bottom limited catenary -2 - catenary with bottom interaction -3 - catenary with no possibility for bottom interaction |
| 7 | KSKP | Node numbering increment for element generation (see Note 4) |
| 8 | ISIGO | Initial state flag (give on first element record only; see Note 4) <ul style="list-style-type: none"> 0 - equilibrium initial state with valid node positions and tensions. 1 - compatible initial state with node positions defining unstretched lengths. 2 - guessed deformed state. Unstretched lengths given in Word (10). Node positions not used to compute initial lengths. |
| 9 | SIG (I) | Initial tension (see Notes 2, 3, 4 and 7) <ul style="list-style-type: none"> KOMP\geq0 total element tension KOMP<0 horizontal tension |
| 10 | DSO (I) | Element length (see Note 2) <ul style="list-style-type: none"> \geq0 unstretched length <0 get lengths from the Nth NGEN record |
| 11 | IT (8, I) | Fluid medium flag (the element is assumed to remain in the original medium at all times). See FLUID record (default is fluid 1). |

NOTES:

1. Elements must be in ascending element number order, and the first and last ones must be input. Any omitted elements will be generated using the KSKP value on the element card following the omitted elements. Elements are incremented by 1; node numbers are incremented by KSKP. The element preceding the omitted elements is used to begin the increments.

For example: if element 1, defined by nodes 6 and 12, preceded the omitted elements and KSKP = 4, then the next element generated would be element 2 defined by nodes 10 and 16.

The generated elements will be assumed to have the same properties as given by the element previous to the omitted ones. KSKP defaults to 1 if needed and not input.

2. The main difference in the input for catenary and truss (simplex) elements is the code type given in IT(5,I) and the need to define the initial state for catenaries. The horizontal tension, unit weight per length, and unstretched length need to be provided or calculated. This can be done by direct input on the ELEM record or by using the NGEN and TENS records. These two input options are discussed below.

Using the ELEM record - the unstretched length is given in Word 10 in this case. Using this data and the weight per unit length from the MATE record, the initial state of the catenary can be computed. When IT(5,I) is negative and Word 10 is zero, then Word 9 is assumed to contain the horizontal component of tension. Using this and the weight per unit length defined in the MATE record, SEADYN will compute the initial state using the same equations as used by NGEN. (i.e. given H and W find the line length, the tangent point on the bottom, and initial tensions).

Using the NGEN and TENS records - The NGEN Record is a node generator and has no function relative to defining elements. However, the unstretched lengths may be obtained from the NGEN record by a cross-reference number provided in Word 10 of the ELEM record. The number in W10 references the Nth node generation record for a catenary line in the NGEN data set (NGEN records with NCAT=0 (straight line node generation) are not counted). A negative sign is used on the number in Word 10 to signal that it is not a length. That will cause the initial state of the catenary element to be computed from the horizontal component of tension and weight per unit length provided on that NGEN record. The TENS record calculates tensions for a line of elements. The tension can be input directly or calculated by using information from the NGEN record.

3. A catenary element is signaled by $IT(5,I) < 0$. The types of catenary elements available are:

IT(5,I) = -1: Bottom limited catenary which never leaves the bottom. The bottom node is always assumed to define a flat bottom that is not penetrated by the sag of the line.

IT(5,I) = -2: General catenary with bottom interaction. Both nodes are capable of general movement. Bottom limited behavior is assumed when either node has a vertical fixity imposed. (see LIM1 and LLOC records)

IT(5,I) = -3: General catenary with no bottom limits. Both nodes are capable of general movement. No bottom limits are imposed (sag allowed) regardless of the fixity of the nodes.

4. The state of stable equilibrium is obtained in flexible structures when the strained positions of all components of the system are such that the internal load distribution supports the applied loads. An equilibrium state of a cable system modeled as a set of discrete elements is described by a consistent set of the following:

- unstretched lengths of all elements
- stretched lengths of all elements
- the position of all nodes which is consistent with the stretched lengths (in fact the stretched lengths are computed from these)
- material properties for all elements
- applied loads for the entire system
- boundary conditions

The solution proceeds as follows: starting with the nodal positions, the stretched lengths are computed. Using the stretched and unstretched lengths, the strains are computed. Using the strains and the material properties, the element internal loads are computed. Using the internal loads and the nodal positions, the nodal loads from internal loads are computed. If these nodal loads balance the applied loads, then the system is in equilibrium.

The nodal positions and element tensions as input may or may not represent an equilibrium configuration (for example, SEADYN has no way of knowing if the initial distances between the nodes represent the element lengths in a stretched or unstretched state). This information is input by setting ISIGO on the first element record.

ISIGO = 0: The nodal positions, element tensions, and material properties are assumed to be a reasonable approximation to the equilibrium state for the first SAO. The nodal positions are assumed to represent stretched lengths. Then the tensions and the material properties are used to compute the strains and finally unstretched lengths.

ISIGO = 1: The nodal positions are assumed to represent unstretched lengths. Since there is no information about the stretched lengths in this data, there is no information about internal loads or equilibrium. Any input for element tensions is simply guessed data that can be used to help get the solution started. The solution process generates successive estimates of nodal positions, which are then used to get stretched lengths, tensions, and equilibrium checks.

ISIGO = 2: Putting together a coherent set of nodal positions which imply the correct unstretched element lengths is sometimes a very tedious (if not impossible) task. With this option unstretched lengths are input directly. The nodal positions are then treated as only a guess of a deformed state of the system. As in the ISIGO = 1 case, the tensions are simply guesses that are used to help the solution get started.

Only the ISIGO = 0 option has the suggestion that an equilibrium state is described. The other two make no pretense of equilibrium.

5. The SEADYN catenary elements use generalized equations that can approximate the combined effect of gravity and drag loading. The drag loading is assumed to be uniform over the length of the element. If a portion of the element lies on the bottom ($IT(5,I) = -1$ or -2), drag loading is only applied to the suspended portion. Each catenary element is assumed to have purely planar response. The plane of the element will change from element to element when drag loads are active. The plane for each element is defined by the direction of gravity and the direction of the drag force. The approximation of a uniform drag load may be very crude for long sagged elements. More fidelity in representing drag responses can be obtained by using more elements. This is an important consideration with $IT(5,I) = -3$ elements which have deep sag.

6. Output for catenary elements is essentially the same as for simplex elements but should be interpreted differently. Tension printed for simplex elements represent average values for the element and are best interpreted as the tension at the midlength of the element. The tension given for a catenary element represents the tension at upper end of the element. Besides the regular tension output, optional output for catenary elements is available to show element connection data, tangent vectors, tension at both ends, and the length of line on the bottom.

Other output associated with the catenary elements is the Element Summary Table which lists the following data: Element Number (ELT NO), Horizontal Tension (H), Line Weight per Unit Length (W), Projected Bottom Length (BOTTOM L), Catenary Span (SPAN) and Height (HEIGHT), and Stretched Line Length (STRETCH L). This data represents the initial state described in the input. If the catenary element type is -3 or -2 and not on the bottom, the bottom length shown will be -1.0 . This data is printed at the end of the input interpretation when IPRO (PROB record) is 1.

7. When a string of elements lies along a catenary curve generated by the NGEN record, the initial tension for each element can be obtained using the TENS record. The initial tension will be taken to be the catenary tension half-way between the two nodes defining the element. SIG(I) is not input if the TENS record is used.

6.2.5 FLOW - Flow-field library definitions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | I | Flow-field number (10 max) |
| 2 | IFLCOD (I) | Flow-field code (see Note 1) >0 Resident function <0 User defined function |
| 3 | FLPAR (1, I) | Optional flow-field parameters (see Notes) |
| 4 | FLPAR (2, I) | |
| | . | |
| | . | |
| | . | |
| 42 | FLPAR (40, I) | |

NOTES:

1. Flow-field codes select a resident function or signal a call to the subroutine USRCUR. Positive flow-field codes refer to resident functions (permanently incorporated into SEADYN), while negative codes refer to user defined subroutines (written, compiled, and loaded into SEADYN by the user). A request for a resident function beyond those currently defined will cause an abort of the run. The parameters are used as appropriate for the resident functions, and they are provided in the calling sequence to the user subroutines.

2. The form for the USRCUR subroutine is defined in Appendix D.

3. The three resident flow-field functions (identified by IFLCOD) at present are:

IFLCOD

1 - uniform velocity field

with: FLPAR

- 1 - X component of flow velocity (LT^{-1})
- 2 - Y component of flow velocity (LT^{-1})
- 3 - Z component of flow velocity (LT^{-1})

IFLCOD

2 - velocity flow field table with up to 10 points of depth vs. velocity. Linear interpolation is used between points.

with: FLPAR

- 1 - first depth (L)
 - 2 - X component of flow velocity (L/T)
 - 3 - Y component of flow velocity (L/T)
 - 4 - Z component of flow velocity (L/T)
 - 5 - second depth (L)
- (etc.) repeat depth, 3 components of velocity for up to 10 depths (Words 3 through 42). Depths must be in increasing numerical order. The values given for depth should be the global coordinate in the vertical direction. Velocities outside the defined depths are assumed zero. No extrapolation is done. The user is cautioned that discontinuities in the velocity field may lead to nonconvergent behavior.

IFLCOD

- 3 - exponentially decaying flow between specified depths

The velocity in the X, Y, and Z directions (or 1, 2, 3, respectively) vs depth(d) is given by:

$$V_j(d) = \text{FLPAR}(J+2) * e^{\frac{d - \text{FLPAR}(1)}{d - \text{FLPAR}(2)}}$$

where:

V_j = velocity at depth d in direction j

FLPAR

- 1 - first vertical coordinate for the decaying field (L)
(the location for the largest value of flow)
- 2 - second vertical coordinate for the decaying field (L)
(the location for the zero value of flow)
- 3 - X component of the decaying flow field (L/T)
- 4 - Y component of the decaying flow field (L/T)
- 5 - Z component of the decaying flow field (L/T)

The rate of decay is controlled by the distance between FLPAR(1), and FLPAR(2). The data given for these two points represent positions in the vertical direction in the global coordinate system. These two values can be chosen without being restricted to be between the surface and bottom if that gets the desired result. The direction of the distance between the two depths may be either down or up. For depths that precede FLPAR(1), the current is assumed equal to the value at FLPAR(1). A zero current is obtained for depths beyond FLPAR(2).

6.2.6 FLUID - Fluid media definitions (used only if IDIR \neq 0 on PROB record). Must precede MATE, BODY and SHIP records.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | FDEPTH (I) | Coordinate at fluid surface |
| 2 | -- | Property code |
| | | 0 - read properties from Words (3) and (4) |
| | | 1 - 1.30×10^{-5} ft ² /sec, 64.0 lb/ft ³ (seawater) |
| | | 2 - 1.68×10^{-4} ft ² /sec, 0.0765 lb/ft ³ (air) |
| 3 | FVISC (I) | Kinematic viscosity (L ² T ⁻¹) |
| 4 | FGAM (I) | Specific weight (FL ⁻³) |

NOTES:

1. Provide one record for each fluid in ascending order. The first fluid given is assumed to extend infinitely below its surface.
2. The FDEPTH for the highest fluid need not be given. It is assumed to extend infinitely above the previous fluid. The minimum input required to define a fluid is a single record with Word 2 used to select fluid properties (seawater or air). This implies the fluid covers the entire space and that the lb-ft-sec units are appropriate.
3. Pressure effects from the accumulated depths are ignored.
4. The present version of the program only allows two fluids.

6.2.7 INVENTORY - Units conversion for component inventory (see Appendix II for the inventory contents).

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | FRCVY | Conversion factor for weights and strengths (Default = 1.0) |
| 2 | FLNVY | Conversion factor for buoy diameters and lengths (Default = 1.0) |
| 3 | FDIVY | Conversion factor for line diameters (Default = 1.0) |
| 4 | INVPNT | Inventory Print Flag 0 - do not print inventory 1 - print all inventory entries |

NOTES:

1. The value obtained from the inventory is multiplied by the corresponding conversion factor to change the units to match those implied by the problem input data.
2. Inventory data will not be used if this record is not input.

6.2.8 LIMIT - Limit set definitions (used only if IDIR \neq 0 on PROB record).

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | LIMIT(I) | Limit number (50 max) |
| 2 | CORDLM(I) | Vertical coordinate of limit (L) |
| 3 | TOLIM(I) | Limit tolerance (L) |
| 4 | RELFAC(I) | Release factor (default = 1.001) |
| 5 | JANCR(I) | Fixity code 0 - buoyant limit (surface bound) with limit imposed only on vertical component 1 - weight limit (bottom bound) with limit imposed only on vertical component 3 - weight limit with all three components held when limit is imposed |
| 6 | BFRICT(1,I) | Bottom friction coefficient for static friction (default: no friction) (See Note 4) |
| 7 | BFRICT(2,I) | Bottom friction coefficient for dynamic friction (default: dynamic = static coefficient) |
| 8 | NBTMDR(I) | Alternate drag function number to be used for elements connected to nodes assigned to this limit when they are on the bottom (default is to use the currently defined function) |
| 9-11 | BOTREF(J,I) | Global X,Y,Z coordinates of a tilted bottom plane reference point (L) (See Note 5) |
| 12-14 | BOTNRM(J,I) | Global coordinates (X,Y,Z) of a vector normal to the tilted plane and pointing away from the bottom (L) or Global coordinates (X,Y,Z) of a second point in the plane (L) (See Note 5) |
| 15-17 | | Global coordinates (X,Y,Z) of a third point in the plane (L) (Optional data given only for the three point definition of the plane (See Note 5)) |

NOTES:

1. This data set defines a table of limit conditions. These are assigned to specific nodes using LLOC, BLOC, or possibly NGEN records (see Note 3 of NGEN record).

2. RELFAC gives the ratio of the vertical components of the line tensions to the external loads at the limited node that must be exceeded to release a limit in a transient (DYN) analysis. External loads include weight/buoyancy of the lines, bodies at the node, and any current and point loads. The release factor is always 1.0 in any static analysis.

3. When a limit tolerance is greater than zero any node entering the limited region will be fixed at the position where it is detected. If the limit tolerance is zero, nodes will only be fixed when they overshoot the limit. All overshoot of the limit conditions will be scaled back to the limit position regardless of the size of the limit tolerance.

It is recommended that the limit tolerance be set at zero unless crude approximations of bottom contact can be accepted. There appears to be little computational penalty for selecting a zero tolerance.

4. When a positive value is given for Word 6 it will cause additional iterations to occur in the solution sequences to estimate the effects of Coulomb friction for the portions of line in contact with the bottom. The value given should be for the friction coefficient (e.g., 0.9). When a constrained node is at the bottom limit, the sum of the external and internal forces at the node is used to estimate the friction effect. When that sum represents a force pushing into the bottom, movement parallel to the bottom is resisted by a force proportional to the magnitude of the normal force. The upper bound of the friction force is equal to the normal force times the static friction coefficient. Its direction is chosen to resist or retard motion. When the upper bound is exceeded by the resultant force parallel to the bottom, the node will slide. When a node is sliding in static analyses, the static coefficient continues to be used. When a node is sliding in a transient analysis, the dynamic friction coefficient is used. When the resultant does not exceed that upper bound, the node is fixed to the bottom. This option will automatically set Word 5 to 3, meaning that when a stick condition is imposed, all movement parallel to the bottom is restrained. See Section 7.1.14 SOLU, Word 20 (IFREEZ) for a solution aid that temporarily overrides the friction iterations to assist in getting convergence on bottom interaction problems.

5. The default assumption (no input for Words 9-17) regarding the bottom limit is that it is horizontal. A sloped bottom may be introduced by specifying a point on the plane and giving the orientation of the plane in terms of a vector normal to the plane. The sense (direction) of that vector will be taken to be positive pointing away from the bottom (i.e., pointing into the fluid). Each limit set input may include a distinct definition of a bottom plane. Two options are provided for input of the normal to the plane. The first option is to give the components of the normal vector as Words 12-14. The second option will calculate the normal vector from the coordinates of three points in the plane. The first of these three points is the reference point, given in Words 9-11. If any of the three points is the origin (0,0,0), that must be given as the reference point. This is because null values for the other Word positions are used to identify the options. Null values in Words 15-17 mean that the normal vector is given in Words 12-14. Values given in Words 15-17 mean that three points in the plane are given and that Words 12-14 and Words 15-17 give the global coordinates of the second and third points.

6.2.9 LLOC - Limit location specification.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | LIMSET | Limit number (50 max; see Word (1) on LIMIT record) |
| 2 | LBEG | First node where limit is applied |
| 3 | LEND | Last node where limit is applied |
| 4 | LINC | Node number increment |

NOTES:

1. No more than 10 line elements can be connected to a limited node.
2. Generation of nodes on a catenary can cause generation of up to two limited nodes per catenary. This situation is described in Note 3 of the NGEN record.

6.2.10 MATERIAL - Material table definitions.

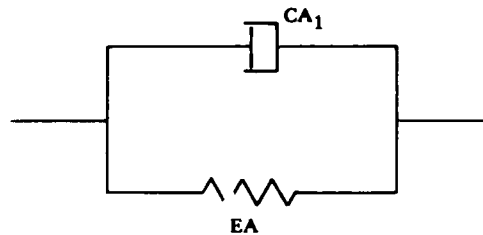
| Word | Variable Name | Description |
|------|------------------|--|
| 1 | I | Material set number |
| 2 | IDRG (I) | Drag function number (see Note 6) 0 - the default drag function will be used (see Appendix C) #0 - the given DRAG function number from the DRAG data set will be used. |
| 3 | DIAM (I) | Cable or line diameter (L). Used only for fluid load computation. |
| 4 | G3 (I) | Weight per unit length (negative means buoyant) (F/L) |
| 5 | MED (I) | Reference medium code. See FLUID record (default is fluid 1). |
| 6 | CAMC (I) | Added mass coefficient (default = 1.0) |
| 7 | TENULT (I) | Ultimate tension capability (F) (see Note 3) |
| 8 | ME (I) | Option flag for tension (T)/strain (ϵ) properties 0 - use exponent form $T = a\epsilon^b$ n - n points in tabular form (maximum n is 20; see Notes 1 and 2) |
| 9 | TT (1, I) | First tension in table or "a" (F) |
| 10 | STR (1, I) | First strain in table or "b" |
| 11 | TTD (I) | Damping parameter CA_1 (FT) (see Note 4) |
| 12 | TTK (I) | Damping parameter EA_1 (F) (see Note 4) |
| 13 | TT (2, I) | Second tension in table (F) |
| 14 | STR (2, I) | Second strain in table |
| | . | |
| | . | |
| | . | |
| 15 | TT (20,I) | (repeat pairs for all table points) |
| 16 | STR (20,I) | |

NOTES:

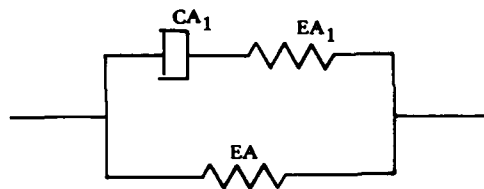
1. When tabular input is used and data are required outside of the given values, the slope of the last (first if only one) segment is assumed to be continued.
2. Input must be in increasing tension order. Thus, materials with compression stiffness must begin by listing the largest compressive load first and progressing to the largest tensile load.
3. The ultimate tension input is used only by the frequency domain dynamic solution. When $TENULT(I)$ is greater than zero, the random estimates for tensions will be factored and compared to the ultimate tension.
4. There are two material damping models provided:

KELVIN Model

EA is provided in the
tension/strain data
 CA_1 is the dashpot coefficient



NOAA-REID Model



EA_1 is an additional stiffness parameter. Additional information regarding the applicability and use of these models can be found in Reference 13.

5. Two methods may be used to describe a material's load-strain information: a two parameter exponential format or a tabular format.

Two Parameter Exponential Format example: $tension = a (strain)^b$

MATE

1, , 0.125, 5, W8, 0, 50000, 1

Here $a = 50000$ and $b = 1$. This means there is a linear relationship between strain and tension with $EA = 50000$.

Tabular Format Example:

Word 8 specifies that 5 sets of load-strain information will be given. Note that Words 13 and 14 are repeated as many times as required to enter the number of data sets specified. The damping terms are zero.

MATE

1, , 0.125, 5, W8, 5, 1000, .01, , , 10000, .06, 25000, 0.1,
(1) (2) (3)

50000, 0.27, 100000, 0.5
(4) (5)

6. Drag functions are defined by the DRAG record. A FLUI record must also be given for the input to be utilized. In the event that a FLUI record is given but no DRAG record is given, then the drag function number, Word (2), refers directly to the drag function code. In that case $IDRG(I) > 0$ refers to resident functions and $IDRG(I) < 0$ refers to user defined functions, given in the USRDRG subroutine. See Appendix D for the definition of USRDRG. This alternate form is provided to retain compatibility with earlier versions of SEADYN. The preferred method is to use the DRAG data record.

6.2.11 NGEN - Generates nodes along a line.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | NUMN | Number of nodes to be generated |
| 2 | NOLD | Beginning reference node (lowest node on a catenary) |
| 3 | I | Ending reference node (highest node on a catenary) |
| 4 | KSKP | Node numbering increment (default = 1) |
| 5 | NBC | Boundary condition flag 0 - do not copy constraint codes. Set all constraints to zero. 1 - copy constraint codes from node NOLD 2 - copy constraint codes from node I |
| 6 | NFIRST | Number of first generated node (default = NOLD + KSKP) |
| 7 | NCAT | Code for line type 0 - straight line 1 - catenary with sag laid flat at limit (default = vertical coordinate of NOLD) 2 - catenary with unconstrained sag 3 - catenary as in 1 except no nodes placed on bottom. The first generated node is located above the bottom tangent point (if any). |
| 8 | GCATW | Catenary weight per unit length (FL^{-1}) |
| 9 | GCATH | Catenary horizontal component of tension (F) |
| 10 | LIMCAT | Limit set number for NCAT = 1 (see Word (1) of LIMIT record). May be used in place of the LLOC data for nodes generated on the bottom by the NCAT = 1 option (see Note 3). |

NOTES:

1. Both the beginning and ending nodes must have been previously defined by NODE or NGEN records. The beginning and ending nodes need have no numerical relation to the generated node numbers; only the spatial relation of the generated nodes is significant.

2. The value of KSKP can be positive or negative. Incrementing starts from NFIRST.

3. All generated nodes are evenly spaced on the generated straight line. Nodes generated on a catenary are placed to give uniform angle change, except for NCAT = 1 with line on the bottom. If line on the bottom is detected with NCAT = 1, the touchdown point is calculated. If the distance between NOLD and the touchdown point is greater than 0.75 times the uniform spacing distance, a node is placed at the touchdown point. If the distance is greater than 1.50 times the uniform spacing distance, NFIRST is placed half-way between the touchdown point and NOLD. The node at the touchdown point is then NFIRST + KSKP, otherwise the touchdown point is NFIRST. (Note: This procedure may result in long element lengths on the bottom adjacent to short elements at the touchdown point. This combination may be troublesome for simplex elements) The remaining generated nodes are placed on a new uniform angle spacing from the touchdown point to the last node. All nodes placed on the bottom are given the limit set number LIMCAT. If LIMCAT is not given, the LLOC (or BLOC) records should be used to set limits.

With NCAT = 3 option the bottom interaction and touchdown point are computed as in NCAT = 1, but no nodes are placed on the bottom. The first generated node is placed according to the uniform angle change from the touchdown point to the top node.

4. The catenary options provided here are simply for the purpose of generating nodes along a catenary line. They have no direct function relative to the generation of catenary line elements (see the ELEM record). Information needed to calculate the unstretched length of a catenary element can be passed through this node generation input. This is done by using NCAT = 3 and following Note 7 of the ELEM record. This information pass-through can be done even when no nodes are to be generated by setting NUMN = 0. This particular situation arises when a single catenary element is desired and the horizontal component of pretension is known instead of the line length.

5. An alternate, and sometimes more simple, option for node generation is available with the NODE data set. See Note 5 of NODE data set.

6.2.12 NODE - Node point definitions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | I | Node number |
| 2 | KSKP | Node generation code (see note 5) |
| 3 | XO (1,I) | X |
| 4 | XO (2,I) | Y |
| 5 | XO (3,I) | Z |
| } | | |
| 6 | NODFIX (1,I) | X |
| 7 | NODFIX (2,I) | Y |
| 8 | NODFIX (3,I) | Z |
| } | | |
| 9 | DFLAG | Angular coordinate flag - "DEGREE" means the nodal coordinates of angle nodes are input in degrees. Otherwise angles are in radians (see Note 1). |
| 10 | XO (4,I) | 1st Euler angle (see Notes 1 and 9) |
| 11 | XO (5,I) | 2nd Euler angle |
| 12 | XO (6,I) | 3rd Euler angle |
| 13 | NODFIX (4,I) | Angular response constraint code (see Note 2) |
| 14 | NODFIX (5,I) | |
| 15 | NODFIX (6,I) | |

NOTES:

1. Each node requires three geometric components to describe its position relative to a global right-handed Cartesian coordinate system. Rigid bodies (ships and mooring buoys) and master nodes use three additional components to describe their orientation in terms of Euler angles (Words 10-12). All angular computations are done in radians, and output for angular position and responses are in radians unless indicated otherwise. Input of initial angular positions can be either in radians or degrees as indicated by DFLAG (Word 9). Only static solutions are able to deal with the six degree of freedom node responses at present. SEADYN contains no operational models for dynamic rigid bodies yet. The solution procedures are capable of processing six degree of freedom responses, but the equations for the rigid bodies must be provided through a custom interface for the time being.

2. Constraint codes define the active and constrained components of movement for each degree of freedom.

- N - means the node is slaved to node N
- 0 - no constraint (free)
- 1 - no movement (fixed), subject to release by limit set conditions. If the component is for a node reserved for payout, this code remains in effect as the node is activated.
- 2 - identifies a component reserved for payout that will be unconstrained (free) when activated. Such a node may be a member of a limit location set and have its constraint codes altered once the node has become activated. Components that are held in reserve for payout cannot be affected by limit constraint activities until they are activated.
- 3 - identifies a component that is unconditionally fixed and cannot be freed by limit constraint activities or payout activation.

A node can have mixed constraint codes. For example, two-dimensional payout nodes could have one constraint code set to 3, and two set to 2.

3. Slave and master nodes may occur in any numerical position in the node numbering scheme. Slave nodes need not have any line elements connected to them. This allows the investigation of the response of specific points on a rigid body at locations other than the attachment of mooring and working lines. Master nodes must have six degrees of freedom assigned to them (see Note 1) since the slave/master relationship requires angular data.

4. A node with any or all constraint codes set to 1, 2, or 3 is still considered an active node even though it cannot move. It is assumed that any of the components of any node can be modified by FREE, FIX, MOVE, LIMIT, or PAYOUT according to the appropriate rules and logic of those options. LIMIT checks involve only members of specific location sets that are not presently held in reserve for PAYOUT or preempted by a MOVE instruction. They will assign a constraint code of 0 or 1 for bottom interaction and 0 or a negative of the limit set number for surface interaction. PAYOUT reserved nodes must have all components assigned either a 1, 2, or 3 fixity code which will be handled as indicated by Note 2. Nodes that are removed from active status by reel-in logic will be assigned fixity codes of 2 if the code while active was 0. Otherwise they will be left alone. The FREE, FIX, and EDIT options allow user intervention in some of these processes. Because of the flexibility provided by some of the input options, the user must become familiar with their features and take care not to violate their logical use.

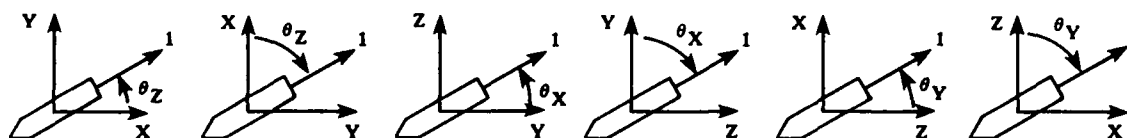
5. Input of nodes need not be in numerical order. Nodes can be entered more than once, and the last entry will be the one used in the solution. Omitted nodes can be generated in a straight line with uniform spacing with the aid of the KSKP parameter. If $KSKP \neq 0$ on a node record, then that node is designated as the last node on the line, and the one input preceding it is designated as the first node on the line. Nodes are generated evenly along the line between these two points with node

numbers incremented by KSKP from the first to the last node. The difference between the node numbers at the ends of the line must be an integer multiple of KSKP, and KSKP cannot be negative. The generated nodes will have the same constraint codes as the first node on the line. More general node generation schemes are available using the NGEN record.

6. All of the NN nodes must be accounted for in the combined specifications for nodes and generation schemes.

7. All points of line attachment to rigid bodies, such as ships or mooring buoys, must be slaved to the primary node defining the body. The program imposes no limit to the number or relative locations of these attachments, except that the mooring buoy rigid bodies use the first two attachments to define the local coordinate system for the body. These two attachments are also used for reference in the CHEK option (see Note 2 on BLOC record).

8. The initial orientation of a ship or rigid body relative to the global coordinate system (see PROB record, Word 3) must be input. The azimuth or heading angle required in this definition can be seen in the following sketch. (The global axes are X, Y, Z and the local ships axes are 1, 2, 3, representing aft, starboard, and up, respectively.)



9. More general position definitions for the master nodes (those having 6 degrees of freedom) require full specification of the Euler angles relating the local roll, pitch and yaw reference coordinate system to the global cartesian X, Y, Z system. All load components (moments) and constraints (node fixities) are assumed to be in the local roll, pitch, yaw (or 1, 2, 3) system. The Euler angles for the transformation are defined relative to the vertical axis in the global system. Their values are for the roll angle relative to the local system, the elevation angle, and the azimuth angle. Thus, the first value is the roll Euler angle, the second is the elevation Euler angle, and the third is the azimuth Euler angle. An easy way of determining the sign convention is to visualize the steps in mapping from the global system to the local system. That mapping starts with a twist about the global vertical axis through a plus azimuth angle. Plus means a positive (right hand rule) twist about the global vertical axis (see the sketch above). This will bring the mapped position of the 1st horizontal axis to be aligned with the horizontal projection of the local 1 or roll axis. Rotation through the elevation angle about the second horizontal axis will rotate the first horizontal axis to be aligned with the local 1 axis. Again, a positive elevation angle represents a positive twist about the 2nd horizontal axis. The final rotation is a twist about the local 1 axis to bring the other two axes to their final positions. Thus, a mapping from global to local is a + azimuth twist about global vertical axis, then a + elevation twist about the intermediate (2) axis, and a + roll twist about the local X or (1) axis. Mapping from the local to the global system is just the reverse.

6.2.13 PAYOUT - Payout/reel-in data.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | I | Payout set number (5 max) |
| 2 | JOP (I) | Initial Node for payout |
| 3 | JPELT (I) | Initial element number for payout |
| 4 | AMAXL (I) | Mitosis length (see Notes) (L) |
| 5 | NGROW (I) | Number of elements available for growth |
| 6 | NSHRNK (I) | Number of elements available for reel-in |
| 7 | NELPOI (I) | Element number increment (default = 1) (see Note 5) |
| 8 | MITNOT (I) | Reference length option flag (see Note 1) = 0 use initial length of element as reference length ≠ 0 use AMAXL(I) as reference length for mitosis |

NOTES:

1. MITNOT is not a length! It is an integer flag which tells how to compute the length. Payout/reel-in is approximated by incrementally changing the unstrained element length. When the unstrained length exceeds AMAXL(I) plus a reference length during payout and NGROW(I) > 0, then a mitosis operation will be performed. The payout element will be divided into two elements. The original payout node will be assigned to the division point, and the new node introduced in the new element will become the payout node. The new element number is obtained by adding NELPOI(I) and JPELT(I). During reel-in the reverse process will occur when the unstrained length is less than AMAXL(I) and NSHRINK(I) > 0. In this case the new element is identified by subtracting NELPOI(I) from JPELT(I). AMAXL(I) = 0.0 causes the mitosis check to be ignored.

The reference length for payout mitosis is either:

The initial unstretched length of the active payout element JPELT(I)

- if the next element to be activated has a different material number
- or if this is the first mitosis for that payout end and MITNOT(I) = 0

or

AMAXL(I) for all other situations.

2. The elements available for payout are inactive until a mitosis activates them by assigning a length to them. These elements must be included in the total number of elements, and their nodes, material, etc., must be defined in the PROBLEM block. The nodes for these inactive elements must be defined with their coordinates having the same initial position as the payout node (causes the element length to be zero). These nodes must be given constraint codes for all degrees-of-freedom. The appropriate codes are:

- 1 - if the component is to remain fixed and subject to limit checks (if any) after mitosis
- 2 - If the component is to be free after mitosis but possibly subject to subsequent limit checks
- 3 - if the component is to remain fixed and not subject to limit checks after mitosis

3. The elements available for reel-in must be active portions of the structure and sequentially connected to the reel-in point (no branches). The nodal constraint codes on the reel-in node will be re-assigned to the new reel-in node, and the deactivated nodes will be appropriately constrained automatically.

4. Payout/reel-in can occur only at fixed nodes or nodes defined in MOVE data sets. In either case, the displacements of all three components must be defined.

5. A positive element number increment means that on PAYOUT the next element activated will have a higher element number than the one currently being lengthened.

6.2.14 SHIP - Ship data definitions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | ISHIP (I) | Node where ship is located (see Note 1) |
| 2 | LSHP (I) | Load function option (see Note 5) -1 - search file on Tape 10 for equivalent ship 0 - use default analytical load functions (Appendix G) N - use N th ship from ship load file (Tape 10). |
| 3 | CPROP (I) | Propeller resistance coefficient (default = 1.00) |
| 4 | CR (I) | Longitudinal hull resistance coefficient only for analytical functions (default = program calculates one, see Note 6) |
| 5 | CS (I) | Hull wetted surface coefficient only for analytical functions (default = 2.70) |
| 6 | CMS | Amidships coefficient to calculate C_R for analytical functions (default = 0.98) |
| 7 | SLT (I) | Total length of ship (L) |
| 8 | SAE (I) | End projected wind area (L^2) |
| 9 | SAS (I) | Side projected wind area (L^2) |
| 10 | SLWL (I) | Water-line length (L) |
| 11 | SBEAM (I) | Beam at midships (L) |
| 12 | SDRFT (I) | Draft at midships (L) |
| 13 | SDSP (I) | Volume displacement (L^3) |
| 14 | APROP (I) | Propeller projected area (L^2) |
| 15 | FSFRW (I) | Load table wind force conversion factor (default = 1.0) |
| 16 | FSFRC (I) | Load table current force conversion factor (default = 1.0) |
| 17 | FSLEN (I) | Load table length conversion factor (default = 1.0) |

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 18 | FSVEL (I) | Load table velocity conversion factor (default = 1.0) |
| 19 | SHIPK (1,I) | Heave restoring coefficient (default = 1.E22 and displacement fixed) |
| 20 | SHIPK (2,I) | Roll restoring coefficient (default = 0.0) |
| 21 | SHIPK (3,I) | Pitch restoring coefficient (default = 0.0) |
| 2 | SHIPK (4,I) | Heave/pitch restoring coefficient (default = 0) |
| 23 | WDEPTH (I) | Water depth at ship location (default = 10 times draft, see Note 7) |

NOTES:

1. Ships are special purpose rigid bodies. The node defining the location of a ship must be a master node with six degrees of freedom. Any number of slave nodes can be assigned to the master node to represent line attachment points, locations of special load application, or reference points on the rigid body where response data is desired. The Euler angle data for the master node must be consistent with the local system discussed in Note 4 (see also the notes for the NODE data set). Linearized ship restoring coefficients can be input if desired. Otherwise, the ship will be assumed fixed in heave during static analyses. The user is responsible for assigning fixity to roll and pitch (if needed) when no restoring coefficient input is given. There are no dynamic equations in SEADYN for ships that will allow transient, mode, or frequency domain solutions to be processed.
2. Words (3) through (14) may be omitted if LSHP(I) = N and no similarity scaling is required.
3. The conversions factors, Words (15) through (18), multiply the values of force, length, and velocity from the ship loading file to get values consistent with the units implied by the rest of the input data.
4. The local coordinate system for ships assumes the 1, 2, and 3 directions are aft, starboard, and up, respectively. This is consistent with the loading conventions of Appendixes A, F, and G.
5. If LSHP(I) = -1, the ship load data files saved on Tape 10 will be searched to find a vessel that most closely matches the parameters given in Words (3) to (14). The equivalent ship load data will be scaled for the ship described in Words (3) to (14) using the procedure in Appendix F.
6. When analytical loading functions are used, the longitudinal resistance coefficient can be input or calculated using a table look up. The total coefficient, CR(I), consists of three parts:

C_r = residuary coefficient

C_f = frictional resistance coefficient

ΔC_f = fouling resistance coefficient

The value for C_f is always calculated in the program using:

$$C_f = \frac{0.456}{(\log_{10} R_e)^{2.58}} - \frac{1700}{R_e} \quad R_e \geq 5 \times 10^5$$

$$C_f = 0.002 \quad R_e < 5 \times 10^5$$

where R_e is Reynold's Number based on the longitudinal component of the velocity. When CR(I) is input, it is taken to be $C_r + \Delta C_f$. When it is zero or no input is given, then C_r is obtained from a table and ΔC_f is assumed to be 0.0005.

7. The water depth is used only in the similarity scaling from tabular ship load data (see Appendix C).

6.2.15 STRUM - Strum string definitions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | ISTRNG (1) | Number of elements in string (max 20) |
| 2 | KSTRNG (1, 1) | List of string elements |
| 3 | KSTRNG (2, 1) | Must be listed in sequence from one end of the string to the other. It does not matter which end is listed first. |
| | . | |
| | . | |
| | . | |
| 31 | KSTRNS (30,1) | |

NOTES:

1. A string consists of adjacent (contiguous) elements to be considered in estimating drag amplification factors for strumming analyses. An element can be contained in more than one strum string definition. The drag amplification factor for each element listed in a string is the largest value it has in any string.

2. The maximum number of strings is 30. The maximum number of elements per string is 20.

3. Drag amplification estimates are made on LIVE, DYN, and TSSS subanalysis with nonzero relative fluid velocities whenever these strum strings are defined. The procedure assumes that the strings are supported at each end and an auxiliary vibration mode analysis is conducted on each string to estimate its possible involvement with vortex shedding-induced strumming. The general approach of Skop, Griffin and Ramberg (Ref 14) is used. The occurrence of the drag amplification estimates is controlled by the changes in relative velocity and the CEPS parameter input on the SOLU record associated with the SAO data set.

6.2.16 TENSION - Initial tension input.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | LBEG | Beginning element number |
| 2 | LEND | Ending element number |
| 3 | LINC | Element increment |
| 4 | LCODE | Option code |
| | | 0 - assigned initial tension, the value in Word (5) |
| | | N - estimate initial tension from the N th catenary defined on the NGEN records (see Note 2) |
| 5 | SIG | Tension to be assigned to elements (F). |

NOTES:

1. Catenary line tensions (LCODE = N) for the Nth catenary are calculated at the midpoint of each element (see NGEN record and ELEMENT record, Notes 3 and 7).
2. The value of N refers to the Nth NGEN record which has an NCAT value other than zero. Straight line generations are not included in the count.

6.2.17 TFUNCTION - Time-function library definitions

| Word | Variable Name | Description |
|------|---------------|--|
| 1 | I | Time-function number (20 max) |
| 2 | ITFCOD (I) | Time-function code (see Notes) <div style="margin-left: 40px;"> >0 Resident Function <0 User-Defined Function </div> |
| 3 | TPARM (1, I) | Optional time-function parameters |
| 4 | TPARM (2, I) | |
| . | . | |
| . | . | |
| 22 | TPARM (20, I) | |

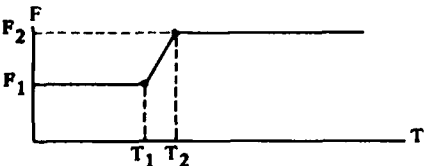
NOTES:

1. Time-function codes select a particular form of resident function or signal a call to the subroutine USRTFN. Positive-function codes refer to resident functions, while negative codes refer to user-defined functions. A request for a resident function beyond those currently defined will cause an abort of the run. The parameters are used as appropriate for the resident functions, and they are provided in the calling sequence to the user subroutine.

2. The USRTFN subroutine is defined in Appendix F.

3. The resident functions presently available:

ITFCOD

| | | TPARM INPUT | | | | | | |
|---|---|----------------|----------------|----------------|----------------|----------|----------|----------|
| 1 | <u>RAMP BUILD-UP/DECAY</u> | <u>1</u> | <u>2</u> | <u>3</u> | <u>4</u> | <u>5</u> | <u>6</u> | <u>7</u> |
| |  | T ₁ | F ₁ | T ₂ | F ₂ | J | 0 | 0 |

J=0 means use a sharp corner transition (as shown in the sketch)

J=1 means use a cosine blend on the transitions instead of the sharp corners

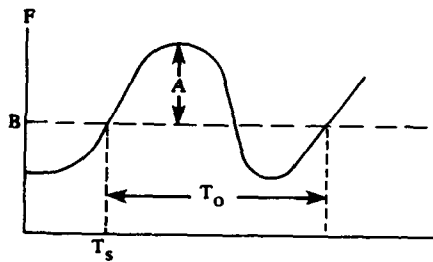
ITFCOD

2 SINE FUNCTION (3 variations are possible)

$$F = A \sin [\omega (T - T_s)] + B$$

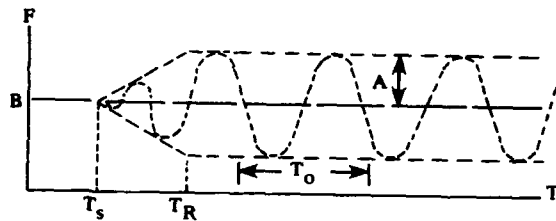
$$\omega = \frac{2\pi}{T_o}$$

SHIFTED SINE



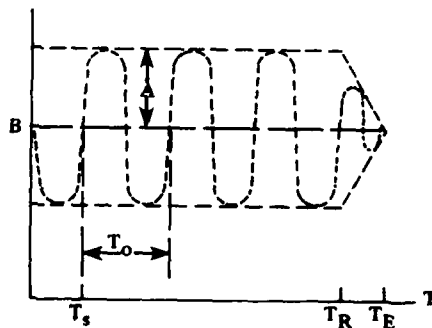
| TPARM INPUT | | | | | | |
|-------------|----------------|----------------|----------|----------|----------|----------|
| <u>1</u> | <u>2</u> | <u>3</u> | <u>4</u> | <u>5</u> | <u>6</u> | <u>7</u> |
| A | T ₀ | T _s | B | 0 | 0 | 0 |

RAMPED SINE



| | | | | | | |
|---|----------------|----------------|---|----------------|---|---|
| A | T ₀ | T _s | B | T _R | 0 | 0 |
|---|----------------|----------------|---|----------------|---|---|

DECAYED SINE



| | | | | | | |
|---|----------------|----------------|---|----------------|----------------|---|
| A | T ₀ | T _s | B | T _R | T _E | 0 |
|---|----------------|----------------|---|----------------|----------------|---|

ITFCOD

3 TABULAR INPUT (max of 9 points plus end values)

TPARM

| | | |
|----|-------|---|
| 1 | F_0 | Function value for times less than T_1 |
| 2 | T_1 | First time point |
| 3 | F_1 | First time function value |
| 4 | T_2 | etc. |
| 5 | F_2 | |
| . | . | |
| . | . | |
| . | . | |
| 18 | T_9 | |
| 19 | F_9 | |
| 20 | F_B | Function value for times greater than last time |

4 SUPPORT MOVEMENT TIME SEQUENCE GENERATED FROM RAO AND SPECTRUM DATA

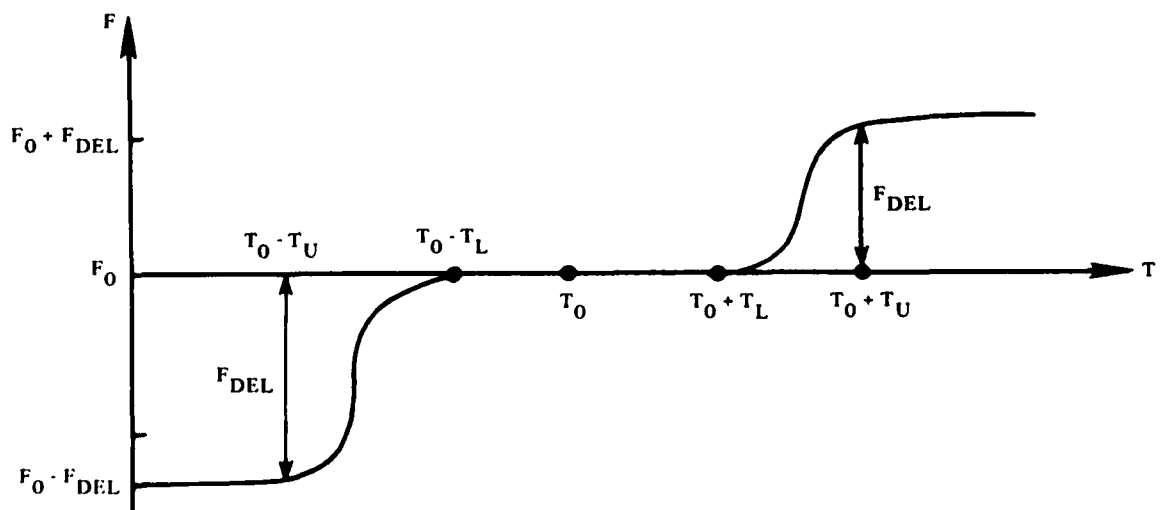
This option generates a table of time sequence data for all three nodal components for the movement of a node. No angular motion data is generated. A pre-generated RAO file must be provided in the data set named 'SCORES.RAO'. A representative time sequence is generated for the x,y,z components of a node in a local coordinate system using the wave spectrum data given here. The local coordinate system data is transformed to the SEADYN global system based on the Euler angle data given here. (See Note 9 of the NODE record for a description of the Euler angles.) No more than 5 time sequences can be defined. These sequences may be produced by multiple selections of ITFCOD = 4 and ITFCOD = 6. The repeated definitions of time sequences with ITFCOD = 4 can select different support points, sea states, and wave spectra but only one RAO file may be used. Linear interpolation is used in the table to get values of the motion between the times defined.

TPARM

| | | |
|---|--------|--|
| 1 | FDT | Time interval for the sequence (default =0.5) |
| 2 | TFINAL | Final time for the sequence (the maximum number of entries in the generated sequence is 500) |
| 3 | NFREQ | Number of frequencies to be used in the generation (default =16, max = 100) |

| | | |
|-----|--------|--|
| 4 | ID | Wave spectrum type code 1 - Neumann 2 - Pierson-Moskowitz 3 - ISSC 4 - (tabular....not supported by this system) 5 - Bretschneider 6 - JONSWAP The spectral ordinates should be scaled such that the integral over frequency equals the mean squared wave amplitude |
| 5 | WD1 | First wave spectrum parameter (consistent with RAO system units and spectrum type) |
| 6 | WD2 | Second wave spectrum parameter (consistent with RAO system units and spectrum type) |
| 7-9 | | Euler angles for the transformation from the model global coordinate system to the support point motion coordinate system (the RAO system). |
| 10 | IPAC | Attachment point number on the RAO file to be processed |
| 11 | ISEAST | Sea state number on the RAO file to be processed |
| 12 | XFAC | Length units conversion factor = $L(\text{on RAO file})/L(\text{in model})$ (default = 1.0) |
| 13 | TFAC | Time units conversion factor = $T(\text{on RAO file})/T(\text{in model})$ (default = 1.0) |

5 DOUBLE COSINE TRANSITION WITH DEAD ZONE



For $T \leq (T_0 - T_U)$

$$F = F_0 - F_{DEL}$$

For $(T_0 - T_U) < T < (T_0 - T_L)$,

$$F = F_0 - .5 * F_{DEL} * (1. - \cos(\pi (T - T_0 + T_L) / (T_L - T_U)))$$

For $(T_0 - T_L) \leq T \leq (T_0 + T_L)$,

$$F = F_0$$

For $(T_0 + T_L) < T < (T_0 + T_U)$,

$$F = F_0 + .5 * F_{DEL} * (1. - \cos(\pi (T - T_0 - T_L) / (T_U - T_L)))$$

For $T > T_0 + T_U$,

$$F = F_0 + F_{DEL}$$

TPARM

1 T_0

2 T_L

3 T_U

4 F_0

5 F_{DEL} (use negative to get reflected form)

6 SUPPORT MOVEMENT SPECIFIED BY AN EXTERNAL DATA FILE

This option generates a table of time sequence data for all three nodal components for the movement of a node. No angular motion data is generated. A pre-generated file must be provided in the data set named 'NODMOV.DAT'. A maximum of 500 points can be extracted from the file to provide a table of the x,y,z motion components of a node in a local coordinate system. The local coordinate system data is transformed to the SEADYN global system based on the Euler angle data given here. (See Note 9 of the NODE record for a description of the Euler angles.) No more than 5 time sequences can be defined. These sequences may be produced by multiple selections of the ITFCOD = 4 and ITFCOD = 6. Only one external file may be accessed to extract the sequences under this option, but multiple extractions may be made. Linear interpolation is used in the table to get values of the motion between the times defined.

TPARM

| | | |
|-----|--------|---|
| 1 | TINITL | Value of time on file for the beginning of the sequence (default = 0.0) |
| 2 | TFINAL | Value of time on file for the last entry in the sequence |
| 3 | TSHIFT | Value of model time at TINITL (default = 0.0) |
| 4-6 | | Euler angles for the transformation from the model global coordinate system to the support point motion coordinate system (the system used in the data file). |
| 7 | XFAC | Length units conversion factor, $L(\text{on file})/L(\text{in model})$ (default = 1.0) |
| 8 | TFAC | Time units conversion factor, $T(\text{on file})/T(\text{in model})$ (default = 1.0) |

6.3 RESTART - Restart data record. Must immediately follow title records when it is used.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|--|--------------------------|---|
| 1 | TYPE | Solution type being restarted (DEAD, LIVE, DYN, NEW). Job will abort if DEAD, LIVE or DYN is given and does not match restart data (see Note 2). |
| 2 | NTAPE | Restart tape number for the data file (1, 2, 3 or 4, see Note 3) |
| 3 | NFILE | The number of the data file to be read from the restart tape. (Default is the last one written) |
| 4 | IDCHK | Identification check flag 0 - No identification check is made 1 - Read the title record from the file and compare the first 10 characters with the 10 characters given in CHKWRD (Word 5); abort if they do not match |
| 5 | CHKWRD | Label check word (see Word 4), 10 characters |
| (The data words beyond this point are not used if Word (1) is NEW) | | |
| 6 | IRST | Restart file flag. (Same function as in SAO data, see SAVE record) |
| 7 | DTRSRT | New restart save time interval |
| 8 | NIPR | New value for number of steps between printing. If this is >0 and not equal to the value given previously, the output interval will be changed by the ratio NIPR (new)/NIPR (old) |
| 9 | DTOUT | New output time interval |
| 10 | IOPOUT | New value for optional output flag |
| 11 | OUTPAR (1) | New values for optional output control parameters (OUTP record, see Note 5) |
| 12 | OUTPAR (2) | |
| 13 | OUTPAR (3) | |

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 14 | OUTPAR (4) | |
| 15 | OUTPAR (5) | |
| 16 | OUTPAR (6) | |
| 17 | DTT | New step size |
| 18 | TMAX | New maximum time (default is old one) |
| 19 | GAMNEW | New integrator parameters (default is old values) |
| 20 | BETNEW | |
| 21 | ALPNEW | |

NOTES:

1. The restart tape referred to by NTAPE and the data file referred to by NFILE must have been created in a previous SAO using the SAVE option. The data file numbers are printed out in the run that saved the restart data.
2. TYPE is used to signal the type of restart to be executed. NEW means the data on the file describes the beginning point of a new SAO set, and an SAO flag record is expected on the next input record. The DEAD, LIVE, and DYN types signal that the SAO data on the file is the same as the SAO form indicated, and that the previous SAO is to be continued using the indicated parameter changes. Subsequent SAO sets can follow after the completion of the restarted SAO. A restart to continue a TSSS SAO should have TYPE = "LIVE".
3. The restart tape can be assigned to any tape unit 1 through 4. If the subsequent computations are to be saved, SEADYN will write on tape unit 1, 2 or 3 depending on the SAO type (see the SAVE record). Unit 4 is provided as an alternate unit to bring in the saved file specified in Word 2 to accommodate operating system constraints.
4. Words 17 through 21 are only used for a time domain restart using DYN, where the previous DYN is to be continued. The parameters allow the user to continue with the original values (the default option) or to modify them individually as needed.

EXAMPLES

Assume that on a previous analysis the DEAD had executed successfully but the LIVE that followed was prematurely aborted by the system after saving six restart files. Now it is desired to restart that LIVE asking for more output over the last step(s). To complete this LIVE, reinitiate the job using the following input:

```
Title Record;  
  
REST  
  
LIVE, 4, 5, W10, 3  
  
END
```

Tape 4 was used since the job will continue to save data on Tape 2 if the restarted job continues past a save interval. It is presumed that additional output is triggered by TABL entry 3 in the previous run. Other examples are given in Section 8.0.

Alternatively, a completely new SAO can be initiated from this last file by entering the following:

```
Title Record;  
REST  
NEW, 4  
SAO Card (LIVE, DYN, etc.)  
.  
.  
.  
END
```

This assumes the previous results were copied to TAPE4. The last file on that tape is used to begin the subsequent SAO.

6.4 TABLE - Optional Output Table Definition. This data record may occur within the problem data block, immediately following the restart data set, and in any SAO data set. It may be used to define or redefine the TABL entries as often as needed.

| <u>Word</u> | <u>Variable</u> | |
|-------------|-----------------|--|
| | <u>Name</u> | <u>Description</u> |
| 1 | I | Table set number (20 max) |
| 2 | NODBEG | Beginning node for this output (default = first node) |
| 3 | NODEND | Ending node for this output (default = last node) |
| 4 | NODINC | Node number increment (default = 1) |
| 5 | ILTBEG | Beginning element for this output (default = first) |
| 6 | ILTEND | Ending element for this output (default = last) |
| 7 | ILTINC | Element increment (default = 1) |
| 8-10 | | (Reserved for future options) |
| 11-100 | IOTABL | A list of the code numbers for the optional output desired. The list may be of arbitrary length and need not be in any numeric order. See note 2 for the descriptions of the code numbers. |

NOTES:

1. The data contained in TABL is referenced by the SAO procedures to produce a list of output code numbers (a template) at the appropriate steps of the analysis. References to the TABL entry number may be made by the OUTP record for static and time-domain analyses or the RESU record for frequency domain analyses. Any of these optional data items can produce large amounts of output. Some of them can produce extremely large amounts. Great caution is advised in preparing these requests.

2. Code numbers for optional output.

| <u>Code No.</u> | <u>Variable</u> | |
|-----------------|-----------------|-----------------------------|
| | <u>Name</u> | <u>Description</u> |
| | | (NODAL DATA) |
| 1 | XC | Present nodal coordinates |
| 2 | XO | Reference nodal coordinates |
| 3 | U | Present nodal displacements |
| 4 | UD | Present nodal velocities |
| 5 | UDD | Present nodal acceleration |

| <u>Code No.</u> | <u>Variable Name</u> | <u>Description</u> |
|-----------------|----------------------|---|
| 6 | US | Reference nodal displacements |
| 7 | UDDS | Reference nodal accelerations |
| 8 | NODFIX | Node point fixity codes |
| 9 | | Solution increment (has various names) |
| 10 | F | Total combined load components |
| 11 | FG | Gravity load |
| 12 | FP | Point load |
| 13 | F1 | Fluid drag load |
| 14 | DU | Force residual |
| 15 | VF | Fluid velocity components at nodes |
| 16 | VW | Relative velocity components at nodes |
| 17 | NODKEY | Node information array. Identifies memory management data such as internal equation number, number of degrees of freedom, coordinate system codes, master node number, limit set number |
| 18 | | |
| 19 | | (ELEMENT DATA) |
| 20 | A | Drag diameter |
| 21 | DS | Present length |
| 22 | DSO | Initial length |
| 23 | DSR | Reference length |
| 24 | ES | Secant modulus |
| 25 | ET | Tangent modulus |
| 26 | GMA | Residual added mass |
| 27 | SIG | Present tension |
| 28 | SIGR | Reference tension |
| 29 | STN | Present strain |
| 30 | DRGAMP | Present drag amplification factor |
| 31 | TH | Present direction (3 components per element) |
| 32 | THR | Reference direction |
| 33 | TRNSTR | Local to global transformation matrix (3X3) |
| 34 | MEDIUM | Fluid medium code |
| 35 | | |
| 36 | | |
| 37 | | |
| 38 | | |
| 39 | | (BODY DATA) |
| 40 | | Ship loads in local system |
| 41 | | Ship loads in global system |
| 42 | | Ship equation details |
| 43 | | Body/buoy loads in local system |
| 44 | | Body/buoy loads in global system |
| 45 | | Body equation details |
| 46 | | Euler angle status report |
| 47 | | Body local to global transformation data |
| 48 | | Slave/master state data |
| 49 | | Slave/master matrices |

| <u>Code No.</u> | <u>Variable Name</u> | <u>Description</u> |
|-----------------|--------------------------|--|
| (SOLUTION DATA) | | |
| 50 | GM | Global mass matrix (caution: large size) |
| 51 | GK | Global stiffness matrix (caution: large size) |
| 52 | GKS | Element stiffness matrices (caution: large) |
| 53 | | List only diagonal terms of global stiffness matrix |
| 54 | | Subroutine trace...lists names of major subroutines as they are called. |
| 55 | | Residual norm status report |
| 56 | | Displacement/velocity norm status report |
| 57 | | Numerical damping progress data |
| 58 | | Solution progress reports |
| 59 | | Solution progress details |
| 60 | | Compressive strain (slack) reports |
| 61 | | Material damping force summary |
| 62 | | Element tension/strain computation details |
| 63 | | Catenary element summary |
| 64 | | Catenary element details |
| 65 | | Catenary solution details (CATFX2 Data) |
| 66 | | Produce standard output for each iteration |
| 67 | | |
| 68 | | |
| 69 | | |
| 70 | | Payout progress reports |
| 71 | | Payout details |
| 72 | | Surface/bottom status reports |
| 73 | | Surface/bottom details (including bottom friction calculations) |
| 74 | | MOVE nodal motion data |
| 75 | | Element drag load computation details |
| 76 | | Local coordinate system data for special nodes (e.g., bottom friction) |
| 77-88 | | Not used |
| 89 | | |
| 90 | | Strum solution details (caution: large) |

6.5 EDIT - Edit certain of the previously defined PROBLEM data. This keyword record and data set may be placed preceeding any of the SAO data sets or immediately following a RESTART that has a NEW option selected.

NOTES:

1. The data records that follow this keyword may be any of the following PROBLEM data records:

DRAG - Define drag data sets

FLOW - Flow-field library definitions

LIMI - Limit set definitions

LLOC - Limit locations

STRU - Strum string definitions

TABI - Define optional output table entries

TFUN - Time function library definitions

Input given in these records will replace or add to the data already defined. Selection of any of the other PROBLEM data records could result in a violation of topology and solution consistency restrictions and will result in an abort of the run.

2. If the changes introduced by the LIMI or LLOC result in a change of constraint status for the current configuration, the affected nodes will have their fixity code altered to the indicated value with the assumption that subsequent solution steps will adjust for the change. For example, suppose node 14 was assigned to limit set 3 originally and it was at that limit at the end of the solution SAO just prior to the EDIT. If that EDIT changed either the definition of limit set 3, removed node 14 from a location set which selected limit 3, or if node 14 was assigned to a different location set that did not have the same constraint for node 14, then the constraint codes for node 14 would be set free. In that case node 14 would be able to move without the original constraint. Unless such a change is done with caution, it may drive the next solution step into some severe, if not unstable, solution iterations.

7.0 SUBANALYSIS OPTION (SAO) DATA

Each subanalysis is headed by an SAO flag record:

| | | |
|-----|------|--|
| 7.1 | DEAD | Nonlinear static analysis with gravity, buoyancy, and point loads |
| | LIVE | Nonlinear static analysis with arbitrary combined loads |
| | TSSS | Time-sequenced static solutions (approximate dynamic analysis using LIVE that allows moved nodes, payout, and other time-varying loads but neglects inertia effects) |
| | DYN | Transient nonlinear dynamic analysis (time domain) |
| 7.2 | MODE | Determination of natural frequencies and mode shapes for current position |
| 7.3 | CHEK | Component adequacy checks |
| 7.4 | END | Run termination |
| | NEW | Ends current problem and begins a new problem -- title card read next |

An SAO flag record has the flag name in Word (1) with no other data in the record. SAO data sets are grouped following the SAO flag to identify solution characteristics, loading, boundary conditions, and output requests. Appropriate default values are assumed when no data are given. Unless explicitly stated, the data records have no required order, since each data set is identified by a keyword.

7.1 DEAD, LIVE, TSSS, DYN Data Set

The data set description for DEAD, LIVE, TSSS, and DYN are given in alphabetical order of the keyword:

Keyword

| | |
|-------|--|
| CURR | Flow-field specification |
| FIX | Applies temporary fixed conditions to nodes |
| FREE | Releases node Fix |
| IMPA | Impacting body input |
| INIT | Describes dynamic initial conditions |
| KEEP | Retains the data defined in the preceding SAO for this SAO |
| LOAD | Specifies load conditions |
| LVAR | Specifies load variation codes |
| MOVE | Specifies displacement/velocity/acceleration at nodes |
| OUTP | Selects output data |
| PAYO | Defines payout/reel-in of lines |
| SAVE | Defines restart file save intervals |
| SBUOY | Specifies vertical surface motion for a body |
| SOLU | Solution option characteristics |
| STEP | Solution step size data |
| SURF | Surface current data |
| TIME | Time step data |
| WIND | Surface wind data |

7.1.1 CURRENT - Flow-field specification.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "CURR" |
| 2 | NFLUID(1) | Flow field number from the library defined in the FLOW record (see NOTES 1 and 2) |
| 3 | CURMUL(1) | Flow field scale factor (multiplier for the values obtained from the selected flow library function) (default=1.0) |
| 4 | NFLVRY(1) | Flow field variation code (see Note 3) |
| 5 | NFLUID(2) | See Notes 1 and 2; repeat NFLUID, CURMUL, NFLVRY for each Flow field to be combined, up to a total of 5 fields. |
| 6 | CURMUL(2) | |
| 7 | NFLVRY(2) | |
| . | . | |
| . | . | |
| 16 | NFLVRY(5) | |

NOTES:

1. Up to 5 fluid flow fields that have been defined in the FLOW data records of the PROB data set may be selected and linearly combined by this option. This linear combination will take values for each flow function selected from the library, multiply by the indicated multiplier (based on the value of CURMUL(I) and the variation function selected by NFLVRY(I)), and perform a vector addition of the resulting flow components for each of the selected flow functions. This sum will be evaluated for the position of each node. For LIVE SAOs the full value of the flow field scale factor is used as the multiplier in summing the velocity components, and the variation code is used to multiply the values of the resulting fluid load.

2. When the value of NFLUID(1) (Word 2) is negative, the flow field will be evaluated only at the beginning of the SAO, and the flow vector values assigned to each node will not be changed during the SAO. Otherwise, the values of the flow vector at each node will be computed for each iteration in the solution.

3. The flow field variation codes are:

For LIVE SAO

- 1 - increase flow field effects incrementally
- 0 - hold flow field at full amplitude
- 1 - decrease flow field incrementally from initial value

For DYN and TSSS SAO

- 0 - hold flow amplitude constant or use the time variation implied by th USRCUR subroutine when it is selected
- N>0 - use the Nth time variation function from the TFUN data set

7.1.2 FIX - Applies fixed conditions to nodes.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "FIX" |
| 2 | ICODE | Constraint code (default = 1, see Note 2) |
| 3 | | List of node component codes CODE = (NODE*10) + I where I = 1, 2, or 3 for the X, Y or Z position, components or I = 4, 5, or 6 for the local angular components |
| 4 (etc.) | | Maximum list length is 98 (use additional FIX records to get more) |

NOTES:

1. FIX causes the selected constraint code to be placed in the constraint code array (NODFIX) for the indicated node component. The node component will remain fixed until released by a FREE record or a LIMIT condition.

2. Allowable constraint codes:

- 1 - Component fixed unless limit condition overrides.
- 2 - Component fixed and reserved for payout. It will become free when activated by a payout mitosis* (see PAYO record). It cannot be released by a limit condition.
- 3 - Component fixed and cannot be released by a limit condition.

3. The MOVE record will override any constraint code.

4. CAUTION: Do not FIX any components of a slave node.

*Mitosis refers to the dividing of an element into two new separate elements; the original element length is replaced by two elements (see Reference 1 and the PAYO data sets for further details).

7.1.3 **FREE** - Removes fixed conditions for nodes.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "FREE" |
| 2 | | List of node component codes CODE = (NODE*10) + I where I = 1, 2 or 3 for X, Y, or Z direction, respectively or I = 4, 5, or 6 for the local angular components |
| 3 | | Maximum list length is 99 (use additional FREE records to get more). |

NOTES:

1. FREE causes a 0 to be placed in the constraint code array (NODFIX) for the indicated component. The component will remain free until reset by a FIX record or a LIMIT condition.
2. CAUTION: Do not FREE any components of a slave node.

7.1.4 IMPACT - Impacting body.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "IMPA" |
| 2 | IMPNOB | Node where impact occurs |
| 3 | IMPBOD | Body table number (see BODY record) |
| 4 | VIB (1) | $\left. \begin{array}{c} V_x \\ V_y \\ V_z \end{array} \right\} \text{Components of body velocity prior to impact}$ |
| 5 | VIB (2) | |
| 6 | VIB (3) | |
| 7 | IOPT | Weight option |
| | | 0 - floating body, no weight added, only adds inertia (e.g., iceberg or snagged vessel) |
| | | 1 - new lumped body added to the system |

7.1.5 INITIAL - Describes dynamic initial conditions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "INIT" |
| 2 | NBEG | Beginning node number |
| 3 | NEND | Ending node number |
| 4 | NINC | Node number increment |
| 5 | VI (1) | VX } components of nodal position velocity |
| 6 | VI (2) | |
| 7 | VI (3) | |
| 8 | VI (4) | Theta (dot) X } components of angular velocity |
| 9 | VI (5) | |
| 10 | VI (6) | |

NOTES:

1. When Word 2 is not zero, the velocity components are assigned only to the individual nodes represented by the values in Word 2 through Word 4. Repeat the INIT records as many times as needed to define all nodal velocities.

2. If NBEG, Word 2, is zero or is not given, the velocity components are assumed to be on all nodes in the system. If this is the last INIT record encountered it will be the only one that will have an effect.

3. After an INIT record with NBEG=0 has been given, individual nodal values can still be changed by INIT records with explicit node numbers input.

7.1.6 **KEEP** - Retains the data from the proceeding SAO.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--------------------|
| 1 | OPTION | "KEEP" |

NOTES:

1. This flag causes the data initialization to be bypassed, thereby allowing the preceding SAO data to remain in effect. Only those data to be changed need be entered.
2. KEEP must be the first flag encountered in the SAO data set if it is used.

7.1.7 LOAD - Specifies load conditions.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "LOAD" |
| 2 | LSET | Load set number (3 max, default = 1) |
| 3 | NBEG | Begin node number |
| 4 | NEND | End node number |
| 5 | NINC | Node number increment (default=1) |
| 6 | NCORD | Coordinate system code (default-global system) (see Note 4) 0 for global system on all components 1 for local system on moments, global on point loads 2 for local system on all components 3 for global system on moments, local on point loads |
| 7 | FP(LST,I) | FX } Point load components, (F) |
| 8 | FP(LSET,I+1) | |
| 9 | FP(LSET,I+2) | |
| 10 | FP(LSET,I+3) | MX } Moment components, (FL) |
| 11 | FP(LSET,I+4) | |
| 12 | FP(LSET,I+5) | |

NOTES:

1. The LVARY record is used to identify the characteristics of each load set (the loads applied, held constant, or removed).
2. Load components are placed in the point load input array, FP, by load set number and node number. NBEG, Word 3, must always be given. The same loads can be assigned to a sequence of nodes by giving the appropriate values for NEND and NINC. When multiple LOAD records assign components of the same node for the same load set, only the last assignment will be used.
3. The moment input is used only for nodes that have been identified as having six degrees of freedom. On any other nodes, the input in Words 10-12 is ignored.

4. For loads specified to be in the global system, the load component orientation is fixed throughout the solution sequence. When a local coordinate flag is set, the loads will be treated as "follower forces" that hold their orientation fixed relative to the local coordinate system defined for the node. Roll, pitch, and yaw moments are an example of "follower forces."

7.1.8 LVARY - Specifies load variation codes.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "LVAR" |
| 2 | ILF (1) | Variation code for load set 1 |
| 3 | ILF (2) | Variation code for load set 2 |
| 4 | ILF (3) | Variation code for load set 3 |
| 5 | JDLD | Gravity load variation code for LIVE subanalysis (see Note 2) |

NOTES:

1. Variation Code:

For DEAD and LIVE

- 1 - increment the load set factor from 0 to 1.0 (apply load)
- 0 - hold the load set factor at 1.0 (steady load)
- 1 - increment the load set factor from 1.0 to 0 (remove load)

For DYN

- 0 - hold the load set factor at 1.0 (steady load)
- I>0 - use the time function number I from the TFUN data set to get the load set factor

2. The only loads used in a DEAD SAO are the internally calculated gravity loads and point loads defined in LOAD record data sets. The gravity loads are assumed to have a +1 variation code in the DEAD SAO. The LIVE SAO assumes a variation code of 0 for gravity loads. When a LIVE SAO has not been preceded by a DEAD SAO it may be advantageous to increment the gravity loads along with the LIVE loads. Setting JDLD to +1 will accomplish this.

7.1.9 MOVE - Specifies node point motion.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "MOVE" |
| 2 | INOD | Node number to be moved (see Note 7 if this node is also a payout point) |
| 3 | MC(1) | X motion code |
| 4 | MVC(1) | X motion variation code |
| 5 | AMP(1) | X motion amplitude |
| 6 | MC(2) | Y motion code (see NOTE 8) |
| 7 | MVC(2) | Y motion variation code |
| 8 | AMP(2) | Y motion amplitude |
| 9 | MC(3) | Z motion code |
| 10 | MVC(3) | Z motion variation code |
| 11 | AMP(3) | Z motion amplitude |

The following data words are used only if the node being moved has six degrees of freedom, i.e., a master node for a rigid body point. (see NOTES 4 and 8)

| | | |
|----|--------|----------------------------------|
| 12 | MC(4) | X rotation motion code |
| 13 | MVC(4) | X rotation motion variation code |
| 14 | AMP(4) | X rotation amplitude |
| 15 | MC(5) | Y rotation motion code |
| 16 | MVC(5) | Y rotation motion variation code |
| 17 | AMP(5) | Y rotation amplitude |
| 18 | MC(6) | Z rotation motion code |
| 19 | MVC(6) | Z rotation motion variation code |
| 20 | AMP(6) | Z rotation amplitude |

The following data words are used only if the value of Word 3 is 4. (See Note 8)

21-23 Euler angles for the transformation from the model global coordinate system to the support point motion coordinate system. (degrees)

NOTES:

1. Motion codes:

- 0 - no motion specified (see Note 4)
- 1 - displacement
- 2 - velocity (DYN and TSSS only)
- 3 - acceleration (DYN and TSSS only)
- 4 - velocity/heading turn option (see NOTE 8)

2. Motion variation codes:

For DEAD, LIVE, MODE, and FREQ

- 1 - increment motion factor from 0 to 1.0 (apply displacement)
- 0 - hold the motion factor at 1.0 (hold displaced position)

The motion factor is the multiplier of the motion amplitude for the increment level (load factor) at the current step.

For DYN and TSSS

- 0 - hold the motion factor at 1.0
- I>0 - use the time function number I from the TFUN data set to get the motion factor (See NOTE 9)

3. The motion amplitude will be an angular displacement, velocity, or acceleration on the angular components (Words 14, 17, and 20) on nodes used to define angular motion of rigid bodies. These components will be relative to the instantaneous position of the local coordinate system for the body.

4. DYN and TSSS require all motion components at a node to be defined. Individual components can have motion defined separately in DEAD, LIVE, and MODE. A zero motion variation code in DYN and TSSS means a fixed component that remains at its position at the beginning of the SAO. A zero motion variation code in DEAD, LIVE, and MODE means there is no constraint on that component.

5. The maximum number of components (X, Y, Z) of imposed displacements in DEAD, LIVE, and MODE is 30. The maximum number of moved nodes in DYN and TSSS is 5.

6. The constraint code array (NODFIX) entry is superseded by a MOVE instruction. The constraint code is restored at the completion of the SAO.

7. When the moved node is also a payout point (see PAYO record), it may be necessary to have the moved node number change as mitosis occurs. In that case it is possible to put an alias in for Word 2. This alias is a negative number that gives the payout set number (Word 1 on PAYOUT card) with a negative sign. The moved node will then always be the current payout node.

8. The specification of a motion code of 4 in Word 3 changes the interpretation of the data words. This option imposes motion in DYN and TSSS that is defined in a special local coordinate system in terms of the velocity and heading at the point. The velocity is assumed to be in a horizontal plane. The data for the velocity of the point is given in the Words for the X motion. The data for heading is given in the Words for the Y motion. It is assumed that the Z motion is the vertical and is uncoupled (specified independently) from the motion in the horizontal plane. The Euler angles must be given for this option to establish the relationship between the global vertical direction and the local vertical (Z) direction which is used to define the heading. The heading angle is a positive twist about the local Z direction. (See Note 9 of the NODE record for a description of the Euler angles.) The motion code given in Word 6 must be one of the following:

- 1 - the motion variation function defines the heading angle
- 2 - the motion variation function defines the heading rate of change

If the moved node has 6 degrees of freedom, this option ignores the input in Words 15 through 23 and assumes roll and pitch are held at their values at the beginning of the SAO.

9. Time functions that define full nodal point motions will over-ride the selection of motion variation codes in the Y and Z directions. These options are the ones that use ITFCOD 4 or 6 in the TFUN definitions. Care must be taken to assure the motion codes on each of the components are consistent with the data in the TFUN table.

7.1.10 OUTPUT - Specifies output data.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "OUTP" |
| 2 | NIPR | Number of solution steps between printing of standard output. (See Notes 1 and 2) |
| 3 | DTOUT | For LIVE with heading changes Angle interval between printing standard output (degrees) For DYN and TSSS (See Notes 3 and 4) Time interval between printing standard output. (T) |
| 4 | IFROUT | Print standard output at the beginning of this step only if this is nonzero |
| 5 | ICTOUT | Print catenary output with standard output if this is nonzero |
| 6 | IOPOUT | Optional output template number. (Cross reference to TABL set number) |
| 7 | IFIOUT | Optional output frequency 0 - only when solution step is completed (converged) 1 - for every iteration up to convergence |
| 8 | OUTPAR(1) | Value of solution parameter where optional output begins (default = start with the first one) >0 - Time (DYN or TSSS) - Load factor between 0.0 and 1.0 (DEAD or LIVE) <0 - Step number |
| 9 | OUTPAR(2) | Value of solution parameter where optional output ends (default = no limit, output continues to end of SA0) |
| 10 | OUTPAR(3) | Increment in solution parameter between printing of optional output. No negative values allowed. (See Note 5, default = do every step) |

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 11 | OUTPAR(4) | Value of heading angle when optional output begins. (default = start with the first one) (degrees) |
| 12 | OUTPAR(5) | Value of heading angle when option output ends. (default = no limit, output continues to end of SAO) (degrees) |
| 13 | OUTPAR(6) | Increment of heading angle between printing of optional output. (default = print for all angle steps) (degrees) |
| 15 | MEPLOT | External Plot Flag (see Note 6) >0 means open a new file if it is not open, write a new header record, and prepare to write <0 means assume file exists, prepare to extend the file with no new header |
| 16 | DTPILOT | Output interval for plot records Time for DYN and TSSS, Load parameter for DEAD, LIVE |
| 17 | IPLTBL(1) | TABL set number defining template for output desired (see Note 7) |
| 18 | IPLTBL(2) | Additional TABL set number |
| 19 | IPLTBL(3) | Additional TABL set number |
| 20 | IOUTBI | Limit status output option flag (see NOTE 8) 0 - no extra output 1 - list changes in constraint status in the standard output 2 - same as 1 plus collect the status output in a data set named 'LIMIT.DAT' -2 - same as 2 but rewind the status output file at the start of the SAO. |
| 30 | | Output flag for GE proprietary interface |

NOTES:

1. The standard output begins with the value of the solution parameter (load factor, heading, time) for the current step. The value of the record counter for restart data file (see SAVE record) is given if appropriate. These are followed by the node point fixity codes, the present node position and velocity for each node, and the tension for each element. If an iterative solution is being used, the number of iterations taken on the step will then be printed. If payout is active in the step, the status of the payout ends will be given. Standard output is given at the end of each subanalysis. The initial position output is always given if IFROUT is nonzero. Output is also given when the load or time parameter exceeds the value of the parameter at the last output plus the output interval. Extra output records can be produced when a restart request does not correspond with an output interval (see SAVE record).
2. NIPR is used to calculate the output interval by multiplying the first step size by NIPR. The output interval remains the same even though the step size is changed unless $NIPR = 1$. When $NIPR = 1$, the output is at every step, regardless of step size.
3. DYN and TSSS can use NIPR or DTOUT to determine the output interval. If both are given, DTOUT has precedence.
4. Output for LIVE SAO with heading changes will occur at angle intervals selected by DTOUT. If DTOUT is not given, the results will be output at every angle calculated. NIPR has no effect on heading change output. DTOUT has no effect on LIVE incremental stages where load magnitude is being varied.
5. Negative values for OUTPAR(1) and (2) signal that they refer to step number, not solution parameter (e.g. load factors, time, heading). The value given for OUTPAR(3) must correspond to the type of data given for OUTPAR(2). If OUTPAR(2) is negative, OUTPAR(3) is the step number increment. If OUTPAR(2) is positive, OUTPAR(3) is the parameter increment.
6. Data in Words 15 through 19 are used to send output to a file suitable for being plotted by an external process. The format of the file is defined in Appendix B.
7. The TABL set number(s) referred to may define the node and element ranges and select any of the node and element data (codes 1-32) except for NODFIX (code 8), the node fixity codes and NODKEY (code 17), the node information array. References to any of the other codes will be ignored.
8. The LIMIT status output will report when the constraint status of any limited node is changed. A simple message stating the node number and the value of the solution parameter (e.g., time) will be given whenever a node is released from the surface or bottom limit. More detailed information is given when a node reaches a limit. This output will occur regardless of the other output options selected.

7.1.11 PAYOUT - Payout/reel-in data.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "PAYO" |
| 2 | I | Payout set number (5 max) |
| 3 | NPOVRY (I) | Time variation function number (see Note 2, default = constant payout rate) |
| 4 | PAYV (I) | Payout rate multiplier for time function (default = 0.0) (LT^{-1}) + for payout - for reel-in |
| 5 | NPTVRY (I) | Tension variation function number (see Note 2, default = no tension variation) |
| 6 | PAYVTN (I) | Payout rate multiplier for tension (default = 0.0) (LT^{-1}) |
| 7 | PAYTEN (I) | Reference tension (default = 0.0) (F) |
| 8 | NPAVRY (I) | Acceleration variation function number (see Note 2, default = no acceleration variation) |
| 9 | PAYVAC (I) | Payout rate multiplier for acceleration (default = 0.0) (LT^{-1}) |
| 10 | PAYACC(I) | Reference acceleration (default = 0.0) (LT^{-2}) |
| 11 | AMAXL(I) | New value for mitosis length (L) (default is to use the previously defined value) (see Note 3) |

NOTES:

1. Payout rate is computed as the sum of three possible effects. These are a predefined time variation, a variation due to current tension in the payout element, and a variation due to the current acceleration of the payout node tangent to the payout element.

2. Variation function numbers refer to time-function numbers defined on the TFUN data set. The input variable to the functions are time, tension and acceleration, as indicated. The time function causes the payout rate to be a function of time, where NPOVRY(I) is the TFUN function number describing the time variation. The tension variation function causes the payout rate to be a function of line tension where a given tension range would result in no payout. Tensions above that range would increase the payout rate, tensions below would cause reel-in. NPTVRY(I) is the TFUN function set number describing the tension variation. Similarly, the acceleration variation function causes the payout rate to be a function of live acceleration; NPAVRY(I) is the TFUN function number describing the live acceleration variation.

3. The purpose of this PAYO record is distinct from the PAYO data in the PROBLEM data set. The latter is used to define the payout topology. That topology cannot be altered. What does change is the status and physical characteristics of the elements in that topology. This PAYO record in the solution data sets activates the payout computations by identifying what payout actions are allowed and driven. Even though the mitosis length is established by the PROBLEM data set, it can be modified at solution time.

When this is done by giving a new mitosis length in Word 11, the MITNOT option (Word 8 of the PROBLEM PAYO data record) is automatically reset to MITNOT = 0. This means that the current unstretched length of the active element becomes the mitosis reference length. Mitosis will not occur until the active element length exceeds the current length plus the new mitosis length. (see NOTE 1 of the PAYO record in the PROBLEM data set.)

7.1.12 **SAVE** - Defines restart file save intervals.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "SAVE" |
| 2 | IRST | Restart file flag 0 - do not save records -N - rewind file and save every N th output record N _{≥1} - extend present file and save every N th output record |
| 3 | DTRSRT | Restart save time interval (DYN and TSSS SAO only) (T) |
| 4 | NIXPRN | Output suppress flag (see Note 3) |

NOTES:

1. Restart files are:

- 01 - DEAD
- 02 - LIVE, TSSS
- 03 - DYN

2. When DTRSRT is given for DYN or TSSS, IRST is used only to signal file rewinds. The restart data is then written when the current time is greater than or equal to the time of the last save plus the save time interval.

3. If the step/time on the SAVE record does not coincide with the OUTPUT record request, then an extra output record will be produced unless NIXPRN ≠ 0.

7.1.13 SBUOY - Specifies vertical motion for a body on the surface.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "SBUO" |
| 2 | NODSUR | Node where body is located |
| 3 | NMOTN (I) | Motion code 1 - displacement 2 - velocity 3 - acceleration |
| 4 | IBS (I) | Time function number (see TFUN record) |
| 5 | SBAMP (I) | Surface motion amplitude (L, L/T or L/T ²) |

NOTES:

1. The motion indicated by these parameters will be imposed on the vertical component whenever the body located at a limit. This means the body node must appear on the LIQC record. When the limit restraint is exceeded, the body is released from the motion and the limit condition.

7.1.14 SOLUTION - Specifies solution option characteristics.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "SOLU" |
| 2 | SOPTN | Solution method: For DEAD, LIVE, TSSS, FREQ VRR (default) MNR RFB For DYN DIM (default) RFB |
| 3 | DMU | Numerical damping factor (see Note 1; MNR default = 0, VRR default = 0.001) |
| 4 | REERR | Residual norm error bound (default = 0.001, see Note 2) |
| 5 | DERR | Displacement and pseudo-velocity norm error bound (default = 0.001, see Note 2) |
| 6 | DALPHA | Proportional damping multiplier of mass matrix (DYN) or Ship angle damping (VRR and MNR) |
| 7 | DBETA | Proportional damping multiplier for stiffness matrix (DYN) or Buoy/Body angle damping (VRR and MNR) |
| 8 | SRCHFC | 1D search factor for MNR method (see Note 4) =0.0 - no 1D search of alternating estimates <0.0 - no alternating estimate checks >0.0 - the 1D search initial guess factor or Alpha integration parameter (VRR default = 1.0) |

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 9 | PARMT | MNR method extrapolation parameter (default = 0.9, see Note 5) or Initial pseudo-time step (VRR default = 1.0) |
| 10 | CEPS | Strum update parameter (default = 0.0001) 0.0 - no drag amplifications for this SAO <0.0 - calculate drag amplifications only once per SAO >0.0 - calculate drag amplification at the beginning of this SAO, and each time the relative velocity norm changes more than CEPS (see Note 6). |
| 11 | | Not used at present |
| 12 | JMPDT | Step-size control number (default = 2, see Note 7) |
| 13 | LMITER | Iteration limit (default = 50 for static, =20 for dynamic) |
| 14 | KONVRT | Number of trials before divergence abort (default = 3) |
| 15 | NRUP | Newton-Raphson update interval. Used on MNR method to signal how often to recalculate the tangent stiffness matrix, K_T . This is a packed word: o Units and tens digits give number of steps before new K_T . o Hundreds and above digits give the number of alternating sign trials on a given step before new K_T (default = 105, i.e., one alternating try and five steps) |
| 16 | ANGLIM | Upper bound on response angle in VRR and MNR iterations (default = 1.0, see Note 8) (degrees) |
| 17 | RNRMLM | Upper bound on initial residual norm for MNR (default = 3.0, see Note 9) |

Freeze flag for bottom friction effects

0 means no freeze

1 means freeze (no sliding allowed) the nodes on the bottom that have friction assigned until bottom interaction iterations have converged, then restart the iterations with the freeze removed

NOTES:

1. The numerical damping factor, DMU, is used to avoid problems with singular or ill-conditioned stiffness matrices. It has no influence on the DIM method. It should be used with caution with the RFB method since it alters the stiffness matrix, hence the equilibrium equations. Possible values are:

| | | | | |
|-------|-----|----|-------|------------|
| | DMU | > | 1.0 | very heavy |
| 1.0 | DMU | >= | 0.1 | heavy |
| 0.1 | DMU | >= | 0.01 | moderate |
| 0.01 | DMU | >= | 0.001 | light |
| 0.001 | DMU | = | | very light |

If $DMU < 10^{-8}$ and a singularity is encountered, then DMU is set to 0.001, and the step is tried again. On a repeated singularity, DMU is increased to 0.1, and the step is tried once more. If the singularity persists, the calculation is aborted. The VRR method actively manages the value of DMU based on solution progress. Changing the initial value changes the nature of the program's progress.

2. The error bounds RERR and DERR are used to test for convergence of the iterations. RERR is used only in the MNR and VRR methods. DERR is used in MNR, VRR, and DIM methods.

The convergence criteria are:

$$\begin{aligned} RNORM &< /RERR/ \\ DNORM &< /DERR/ \end{aligned}$$

where RNORM and DNORM represent norms of the nodal force residual and displacement increments, respectively. Both of these inequalities must be satisfied for convergence of the MNR iterations, while only the displacements are checked in the DIM method. The VRR solution checks the residual (RERR) as well as displacements and velocities (DERR).

The term "norm" is used here to mean a scalar measure of the magnitude of a vector. A norm goes to zero as all of the components of a vector go to zero. Some flexibility in the form of these norms is available. If the value of RERR is greater than zero, then:

where:

$$RNORM = \frac{\left[\sum_{i=1}^N R_i^2 / N \right]^{1/2}}{T_{max}}$$

R_i = the i^{th} component of the nodal force residual

N = the total number of nodal degrees-of-freedom

T_{max} = the maximum element tension

If the value of RERR is negative, then:

where:

$$RNORM = \frac{\left[\sum_{i=1}^N R_i^2 \right]^{1/2}}{\left[\sum_{i=1}^M R_i^2 \right]}$$

M = the total number of nodal components including the fixed nodes; i.e., the reactions are included in the denominator (MNR method only).

If the value of DERR is greater than zero on static analyses, then:

where:

$$DNORM = \left[\sum_{i=1}^N \Delta^2(\Delta q_i) / N \right]^{1/2}$$

$\Delta(\Delta q_i)$ = the i^{th} component of the change in the displacement increment.

If DERR is negative on static analyses (except VRR method), then:

$$DNORM = \frac{\left[\sum_{i=1}^N \Delta^2(\Delta q_i) \right]^{1/2}}{\left[\sum_{i=1}^N \Delta q_i^2 \right]}$$

In the DYN subanalyses, the norm selection is the opposite of that selection used in static analyses. See the discussion of norms in Reference 7.

3. The terms DALPHA and DBETA can be used to specify internal damping proportional to the mass and/or stiffness matrix. The assumed form of the damping matrix is:

$$[C] = \alpha [M] + \beta [K]$$

This procedure can be used in DYN subanalysis.

4. The 1D search factor, SRCHFC, can be used with the MNR method in an attempt to enhance a poor initial configuration. See the discussion of the MNR method in Reference 8.

5. PARMT is an extrapolation parameter used in the incremental MNR method. The starting estimate for the displacement on all but the first step is the accumulated displacements from the previous step plus PARMT times the change in displacement calculated from the preceding step.

6. When strum strings are defined in the STRUM record and CEPS is not zero for LIVE, DYN or TSSS SAO's, drag amplification factors will be computed for the strings. This computation will be done at the beginning of the subanalysis and whenever the relative velocity between the fluid and cable elements changes significantly. A significant change is indicated by:

$$\frac{VNORM_i - VNORM_R}{VNORM_R} \geq CEPS$$

where $VNORM = \left[\frac{\sum_{i=1}^N [VF(I) - UD(I)]^2}{N} \right]^{1/2}$

$R =$ time when drag amplification were last calculated
 $i =$ present time
 $VF =$ fluid velocity components at each node
 $UD =$ nodal velocity components (DYN only)

7. JMPDT is for time domain methods only and will not work for static solution methods. Time domain analyses which make repeated time step changes, should be rerun with the step forced to remain below the range of changes. This is done by specifying DT (TIME record, Word 2) and setting JMPDT to -1.

8. Static analyses involving ships and other rigid bodies can encounter problems in controlling angular responses of the bodies. ANGLIM is used to avoid irrational angle changes in the iterative or incremental process. Whenever the largest angle change in an iteration exceeds ANGLIM, the entire change in response is scaled back to assure the limit is not exceeded. This has the effect of a numerical damping or underrelaxation. Resetting ANGLIM to a small value may prove to be a strong impediment to the progress of the analysis since very little net movement would be allowed on each iteration. Giving a large value for ANGLIM essentially removes its influence.

9. MNR solutions check the value of the residual norm at the beginning of a solution step. If that norm is larger than RNRMLM and there are load variation codes which allow loads to be adjusted, then the load step size is scaled back to the value that will make the residual norm equal to RNRMLM. The usual MNR solution then proceeds under the control specified in this SOLU record.

7.1.15 **STEP** - Solution step-size data (static solutions).

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "STEP" |
| 2 | JSTEPR | Number of steps for load incrementing (default = 1) |
| 3 | NSTUP | Start up option N>1 - divide first step into N substeps in an arithmetic progression |
| 4 | HEDINC | Surface load heading increment (degrees) |
| 5 | HEDEND | Surface load total heading change (degrees) |

NOTES:

1. JSTEPR indicates the number of load/movement steps requested. During this phase of the SAO, the load and movement magnitudes are varied. When HEDINC is not zero, a sequence of static configurations will be generated following the convergence or completion of the magnitude incrementing. This additional loading sequence moves the surface wind and current headings through the excursion defined by HEDEND and the heading variation codes.
2. See Note 9 of the NODE record for global heading conventions.
3. See the SURF record and WIND record for initial headings and variation codes.

7.1.16 SURFACE - Surface current data.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "SURF" |
| 2 | CURNT | Surface current velocity (LT^{-1}) |
| 3 | CAD | Initial current heading (degrees) |
| 4 | KODEC (1) | Variation code for fixed heading (load varies) |
| 5 | KODEC (2) | Variation code for variable heading (load constant) |

NOTES:

1. See Note 9 on the NODE record for global heading convention.
2. See STEP record.
3. Variation codes:

- 0 - hold constant at value given
- 1 - increment down (not for heading variation)
- 1 - increment up

4. Ships will obtain loading coefficients from a data file or built-in functions (see SHIP record and Appendixes A, F, and G). Buoys will use the BWND and BSCD parameters (see BODY record). If the buoy is pulled away from the limit, its loads will be calculated from the CURR record defined flow field and the body drag functions (built-in or user defined). See Word (2) on BODY record.

7.1.17 TIME - Time step data.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "TIME" |
| 2 | DT | Time step (see Note 1) |
| 3 | TMAX | Final time |
| 4 | T | Initial time (see Note 3) |
| 5 | KPDTHL | Time step change flag 0 means allow normal time step changes 1 means do not allow growth of the time step beyond the initial value. Time step decreases will be allowed but subsequent increases will be limited by the initial value of the time step. |
| 6 | ALPNEW | α integration parameter (default = 1.0 for VRR, 0 for MNR and DIM) |
| 7 | BETNEW | β integration parameter (default = 1/12) |
| 8 | GAMNEW | γ integration parameter (default = 1/2) |
| 9 | TDSTUP | Not used at present |
| 10 | ITDSUP | Not used at present |

NOTES:

1. If $DT \leq 0.0$, the time step will be internally estimated. If $DT = -A$, then the estimated value will be multiplied by A . If $DTU = -B$, then DTU is set to B times the DT estimate.
2. The integration parameters are those of the generalized Newmark difference equations.
3. A nonzero initial time can be used for TSSS or DYN SAO to adjust to time in the TFUN records. The default is the value of time at the end of the previous SAO. Time is initialized at zero and is changed only by DYN and TSSS. This means that successive DYN or TSSS SAOs will retain the initial time reference when Word 4 is (recommended) allowed to take its default value.

7.1.18 WIND - Surface wind data.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "WIND" |
| 2 | WIND | Wind velocity (LT^{-1}) |
| 3 | WAD | Initial wind heading (degrees) |
| 4 | KODEW (1) | Variation code for fixed heading (load varies) |
| 5 | KODEW (2) | Variation code for variable heading (load constant) |

NOTES:

1. See Note 9 of the NODE record for global heading convention.
2. See STEP record.
3. Variation codes:
 - 0 - hold constant at value given
 - 1 - increment down (not for heading variation)
 - 1 - increment up

7.2 MODE - Mode shape calculation

The calculation of natural frequencies and mode shapes has only four optional data records: MSOL, FIX, FREE and MOVE. The FIX, FREE, and MOVE records are the same as for DEAD, LIVE, TSSS, FREQ, and DYN. They allow imposition or release of constraints prior to mode calculations. MSOL selects solution options. An MSOL record is not needed if the defaults are acceptable.

7.2.1 MSOL - Specifies solution format parameters.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "MSOL" |
| 2 | MODEI1 | Mode shape order flag 0 - list mode shapes in order of increasing frequency 1 - list in order calculated |
| 3 | MODEI2 | Mode shape output flag 0 - print all mode shapes N - print N mode shapes -N - print N mode shapes and write them on logical unit 20 |
| 4 | IBG | Optional output flag 0 - no extra output >0 - print solution details WARNING: very voluminous output recommended only for debug purposes on small models. |
| 5 | FREQL | Lower bound on frequency output range (T^{-1}) (default is zero) |
| 6 | FREQH | Upper bound on frequency output range (T^{-1}) (default is no limit) |

NOTES:

1. Only results for frequencies between the lower and upper bounds will be output.

7.2.2 FIX - Applies fixed conditions to nodes.

(This record is identical to FIX in Section 7.1.)

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "FIX" |
| 2 | ICODE | Constraint code (default = 1, see Note 2) |
| 3 | | List of node component codes $\text{CODE} = (\text{NODE} * 10) + I$ where $I = 1, 2, \text{ or } 3$ for X, Y or Z direction, respectively or $I = 4, 5, \text{ or } 6$ for the local angular components |
| 4 (etc.) | | Maximum list length is 98 (use additional FIX records to get more) |

NOTES:

1. FIX causes the selected constraint code to be placed in the constraint code array (NODFIX) for the indicated node component. The node component will remain fixed until released by a FREE record or a LIMIT condition.
2. Allowable constraint codes:
 - 1 - Component fixed unless limit condition overrides.
 - 2 - Component fixed and reserved for payout. It will become free when activated by a payout mitosis* (see PAYO record). It cannot be released by a limit condition.
 - 3 - Component fixed and cannot be released by a limit condition.
3. The MOVE record will override any constraint code.
4. CAUTION: Do not FIX any components of a slave node.

*Mitosis refers to the dividing of an element into two new separate elements; the original element length is replaced by two elements (see Reference 1 and the PAYO data sets for further details).

7.2.3 FREE - Removes fixed conditions for nodes.

(This record is identical to FREE in Section 7.1.)

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | OPTION | "FREE" |
| 2 | | List of node component codes CODE = (NODE*10) + I where I = 1, 2 or 3 for X, Y, or Z direction, respectively or I = 4, 5, or 6 for the local angular components |
| 3 | | Maximum list length is 99 (use additional FREE records to get more). |

NOTES:

1. FREE causes a 0 to be placed in the constraint code array (NODFIX) for the indicated component. The component will remain free until reset by a FIX record or a LIMIT condition.
2. CAUTION: Do not FREE any components of a slave node.

7.2.4 MOVE - Specifies node point motion.

(This record is identical to MOVE, Section 7.1.9)

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|--|
| 1 | OPTION | "MOVE" |
| 2 | INOD | Node number to be moved (see Note 7 if this node is also a payout point) |
| 3 | MC(1) | X motion code |
| 4 | MVC(1) | X motion variation code |
| 5 | AMP(1) | X motion amplitude |
| 6 | MC(2) | Y motion code (see NOTE 8) |
| 7 | MVC(2) | Y motion variation code |
| 8 | AMP(2) | Y motion amplitude |
| 9 | MC(3) | Z motion code |
| 10 | MVC(3) | Z motion variation code |
| 11 | AMP(3) | Z motion amplitude |

The following data words are used only if the node being moved has six degrees of freedom, i.e., a master node for a rigid body point. (see NOTES 4 and 8)

| | | |
|----|--------|----------------------------------|
| 12 | MC(4) | X rotation motion code |
| 13 | MVC(4) | X rotation motion variation code |
| 14 | AMP(4) | X rotation amplitude |
| 15 | MC(5) | Y rotation motion code |
| 16 | MVC(5) | Y rotation motion variation code |
| 17 | AMP(5) | Y rotation amplitude |
| 18 | MC(6) | Z rotation motion code |
| 19 | MVC(6) | Z rotation motion variation code |
| 20 | AMP(6) | Z rotation amplitude |

The following data words are used only if the value of Word 3 is 4. (See Note 8)

21-23 Euler angles for the transformation from the model global coordinate system to the support point motion coordinate system. (degrees)

NOTES:

1. Motion codes:

- 0 - no motion specified (see Note 4)
- 1 - displacement
- 2 - velocity (DYN and TSSS only)
- 3 - acceleration (DYN and TSSS only)
- 4 - velocity/heading turn option (see NOTE 8)

2. Motion variation codes:

For DEAD, LIVE, MODE, and FREQ

- 1 - increment motion factor from 0 to 1.0 (apply displacement)
- 0 - hold the motion factor at 1.0 (hold displaced position)

The motion factor is the multiplier of the motion amplitude for the increment level (load factor) at the current step.

For DYN and TSSS

- 0 - hold the motion factor at 1.0
- I>0 - use the time function number I from the TFUN data set to get the motion factor (See NOTE 9)

3. The motion amplitude will be an angular displacement, velocity, or acceleration on the angular components (Words 14, 17, and 20) on nodes used to define angular motion of rigid bodies. These components will be relative to the instantaneous position of the local coordinate system for the body.

4. DYN and TSSS require all motion components at a node to be defined. Individual components can have motion defined separately in DEAD, LIVE, and MODE. A zero motion variation code in DYN and TSSS means a fixed component that remains at its position at the begining of the SAO. A zero motion variation code in DEAD, LIVE, and MODE means there is no constraint on that component.

5. The maximum number of components (X, Y, Z) of imposed displacements in DEAD, LIVE, and MODE is 30. The maximum number of moved nodes in DYN and TSSS is 5.

6. The constraint code array (NODFIX) entry is superseded by a MOVE instruction. The constraint code is restored at the completion of the SAO.

7. When the moved node is also a payout point (see PAYO record), it may be necessary to have the moved node number change as mitosis occurs. In that case it is possible to put an alias in for Word 2. This alias is a negative number that gives the payout set number (Word 1 on PAYOUT card) with a negative sign. The moved node will then always be the current payout node.

8. The specification of a motion code of 4 in Word 3 changes the interpretation of the data words. This option imposes motion in DYN and TSSS that is defined in a special local coordinate system in terms of the velocity and heading at the point. The velocity is assumed to be in a horizontal plane. The data for the velocity of the point is given in the Words for the X motion. The data for heading is given in the Words for the Y motion. It is assumed that the Z motion is the vertical and is uncoupled (specified independently) from the motion in the horizontal plane. The Euler angles must be given for this option to establish the relationship between the global vertical direction and the local vertical (Z) direction which is used to define the heading. The heading angle is a positive twist about the local Z direction. (See Note 9 of the NODE record for a description of the Euler angles.) The motion code given in Word 6 must be one of the following:

- 1 - the motion variation function defines the heading angle
- 2 - the motion variation function defines the heading rate of change

If the moved node has 6 degrees of freedom, this option ignores the input in Words 15 through 23 and assumes roll and pitch are held at their values at the beginning of the SAO.

9. Time functions that define full nodal point motions will over-ride the selection of motion variation codes in the Y and Z directions. These options are the ones that use ITFCOD 4 or 6 in the TFUN definitions. Care must be taken to assure the motion codes on each of the components are consistent with the data in the TFUN table.

7.3 CHEK - Checks component data.

The CHECK SAO set consists of the CHEK flag record, and a string of component selection records.

7.3.1 ANCH, BUOY, LINE - Component requests.

| <u>Word</u> | <u>Variable Name</u> | <u>Description</u> |
|-------------|--------------------------|---|
| 1 | CTYPE | "ANCH", "BUOY", "LINE", "DONE" |
| 2 | NELD | Node or element number |
| 3 | ICODE | Component code |
| 4 | JCODE | Bottom factor code |
| 5 | CAPY | Component capacity (default = use inventory with SAFAC) |
| 6 | CMPID | Component Identifier for Inventory Anchor weight Chain size Line diameter Buoy O.D. (input in inventory units) |
| 7 | SAFAC | Safety factor (default = 1.0) |

NOTES:

1. CTYPE identifies the type of component to be checked. Only anchors, buoys, or lines are allowed. Any number of cards can be given. Input is terminated by CTYPE = "DONE".
2. Negative ICODE with CTYPE = ANCH means the vertical load will be used in the capacity check.
3. The CHEK SAO can be initiated following any other subanalysis. If the previous one was not FREQ, the present state is evaluated. If it was FREQ and the dynamic response file was generated, the dynamic tensions will be included in the check for those elements of the file. (See Note 2 of FSOL record)
4. If CTYPE is "ANCH" or "BUOY", then NELD is the node where the anchor or buoy is located. Anchor checks can be made for fixed nodes as well as for active nodes where limits have been defined.
5. No more than ten lines can be attached to any anchor or buoy at an active node with limits (see Note 3 ILOC record). Anchor checks at fixed nodes allow up to twenty lines to be attached to the fixed node, which has no limit conditions specified.

6. Component codes:

ANCHORS

1. Navy standard stockless with stabilizers
2. NAVSHIPS lightweight
3. NAVFAC STATO
4. Imbedment (no inventory)
5. Stock (Admiralty) (no inventory)
6. Mushroom (no inventory)

BUOYS

1. Bar riser chain
2. Spherical or other (no inventory)

LINES

0. Chain
1. Samson 2-in-1 braided nylon
2. Samson 2-in-1 power braid
3. Samson 2-in-1 stable braid
4. Samson Blue Streak
5. Other (no inventory)

7. Bottom factor codes for anchors:

1. Compacted sand
2. Stiff dense clay
3. Sticky clay of medium density (cohesive)
4. Soft mud (fluid), loose coarse sand, gravel
5. Hard bottom (rock, shale, boulders)

8. Contents of the inventory are presented in Appendix II.

7.4 END, NEW - Problem termination.

The END flag record signals the completion of the SAO sequence with no more data to be read. The run is terminated.

The NEW flag record signals the completion of the SAO sequence. The next data record is a title card to begin a new problem. This title card must use the character string delimiters, and only one title record is allowed.

8.0 EXAMPLE PROBLEMS

This section contains input for four example problems and the description of some optional approaches to solving them. A successful execution occurs when the last line of output has "NORMAL TERMINATION". Although not exhaustive, these problems demonstrate many of the features of SEADYN. The models have purposely been kept small and geometrically simple to make them convenient to handle. The user should be aware that SEADYN is capable of modeling much more complicated geometries.

8.1 Towed Body Example

A spherical body towed at the end of a 280-foot line will be used to demonstrate various methods for starting the problem, the effects of grid coarseness, and changing a quasi-static solution to a dynamic solution. The pertinent problem data are:

Tow Cable Data

Unstretched length 280 ft = 3,360 in.
Drag diameter 0.35 in.
Cable weight (in seawater) 0.169 lb/ft = 0.014 lb/in.
EA 1.92×10^5 lb
Normal drag coefficient, C_N $1.5 + 4.0 (R_e)^{-1/2}$
Tangential Drag Coefficient, C_T $0.02 C_N$

Body Data

Weight (in seawater) 580.9 lb
Effective diameter 1.0 ft = 12 in.

The first part of the problem is to determine the position of the cable and body when towed in a straight line at a constant velocity of 10.5 knots. A convenient model for this is to assume the tow point is fixed and the system is subjected to a uniform current equal to the tow speed. This allows the static solution form to be used.

Although the general shape of the system at the tow speed can be guessed, the exact positions of the nodes and the tensions in the various portions of the cable are not easy to compute. Three approaches to calculating the steady-state towing shape are outlined.

APPROACH 1: Begin by assuming the line is hanging straight down. Apply the gravity load with no current to get the suspended tension distribution. Incrementally apply the current to deploy the cable in an approximate solution (RFB method). Iterate on the deployed state to satisfy equilibrium (MNR method). The input for accomplishing this is shown in Table 8-1. The tow cable drag coefficients are given by the USDRG subroutine shown in Table 8-4. It should be noted that the first step requires artificial tensions to get started. Numerical damping and the MNR solution work well here. The VRR method could also be used.

Table 8-1. TOWED BODY APPROACH 1 INPUT

TOWED BODY DEMONSTRATION PROBLEM

APPROACH 1 --- START FROM VERTICAL;

PROB

11,10, 2,1,386

FLUI

0,0,.00252,.037

BODY

1,,580.9,12

DRAG; 1,-1

MATE

1,1,.35,.014112,1,W8,0,1.92E5,1

NODE

1,,,3360

*BODY NODE

11,1,W6,1,1,1

*TOWED NODE

IMPLIED GENERATION

ELEM

1,1,2,,1,W8,1

*ASSUME UNSTRETCHED LENGTHS

10 10 11,,1

BLOC;1,1

*LOCATE BODY AT NODE 1

FLOW

1,1, 212.8

DEAD

*DEPLOY LINE STRAIGHT DOWN WITH MNR + DAMPING

SOLU,MNR,.001

LIVE

*MOVE TO APPROX TOWED SHAPE WITH 10 STEP RFB

CURR,1,1,,1

SOLU,RFB

STEP,10

LIVE

*ITERATE TO CORRECT STATE WITH VRR

CURR,1

END

APPROACH 2: Begin by assuming the line is deployed horizontally in an unstretched state. Apply the gravity and current simultaneously in increments to get an approximate solution (RFB method). Dummy tensions are assumed. Iterate on the deployed state to satisfy equilibrium (MNR method). The input is shown in Table 8-2. The tow cable drag coefficients are given by the USRDRG subroutine shown in Table 8-4.

Table 8-2. TOWED BODY APPROACH 2

TOWED BODY DEMONSTRATION PROBLEM

```

APPROACH 2  START FROM HORIZONTAL;
PROB;11,10,2,1,386
FLUI;0,0,.00252,.037
BODY;1,0,580.9,12
BLOC;1,1
DRAG; 1,-1
MATE
1,1,.35,.014112,1,W8,0,1.92E5,1
NODE
1,,3360
11,W6,1,1,1
NGEN;9,1,11
ELEM
1,1,2,,1,W8,1          *ASSUME UNSTRETCHED LENGTHS
10,10,11,,1
TENS;1,10,1,,500       *APPLY DUMMY TENSIONS
FLOW;1,1, 212.8
LIVE                    *MOVE TO APPROX STATE WITH RFB
    CURR,1 ,1
    SOLU,RFB
    STEP,100,10
    LVAR,W5,1           *SET GRAVITY LOAD FLAG
LIVE
    SOLU,MNR
    CURR,1
END

```

APPROACH 3: Begin by guessing an angle that approximates the deployed state. Deploy the unstretched cable at that angle. Iterate on this guessed state to satisfy equilibrium. Input for this approach is given in Table 8-3. The tow cable drag coefficients are given by the USDRG subroutine shown in Table 8-4. The VRR method is used in this case since the quality of the initial guess is very low and divergent MNR iterations can be expected. The VRR method automatically compensates for the lack of initial tension.

Table 8-3. TOWED BODY APPROACH 3

TOWED BODY DEMONSTRATION PROBLEM

```

APPROACH 3  START FROM SLOPED LINE;
PROB;11,10, 2,1,386
FLUI;0,0,.00252,.037
BODY;1,0,580.9,12
BLOC;1,1
DRAG; 1,-1
MATE
1,1,.35,.014112,1,W8,0,1.92E5,1
NODE
1,,3195.55,1038.297
11,W6,1,1,1
NGEN;9,1,11
ELEM
1,1,2,,1,W8,1          *ASSUME UNSTRETCHED LENGTHS
10,10,11,,1
FLOW;1,1, 212.8
LIVE                    *ITERATE ON GUESS TO CONVERGENCE WITH VRR
      SOLU,VRR
      CURR,1
END

```

These three approaches are not the only possible options. In each approach, the beginning state was described by the positions of the nodes in an unstretched state (ISIGO = 1). Other approaches can be devised quite easily.

The number of elements for these tests was chosen somewhat arbitrarily. The results of changing the number of elements are summarized in Figure 8-1. The figure shows quite accurate results even for very crude models. These models have assumed uniform spacing of the nodes for input convenience. The equilibrium state suggests it would be more appropriate to put shorter elements near the towed body where the curvature is the greatest.

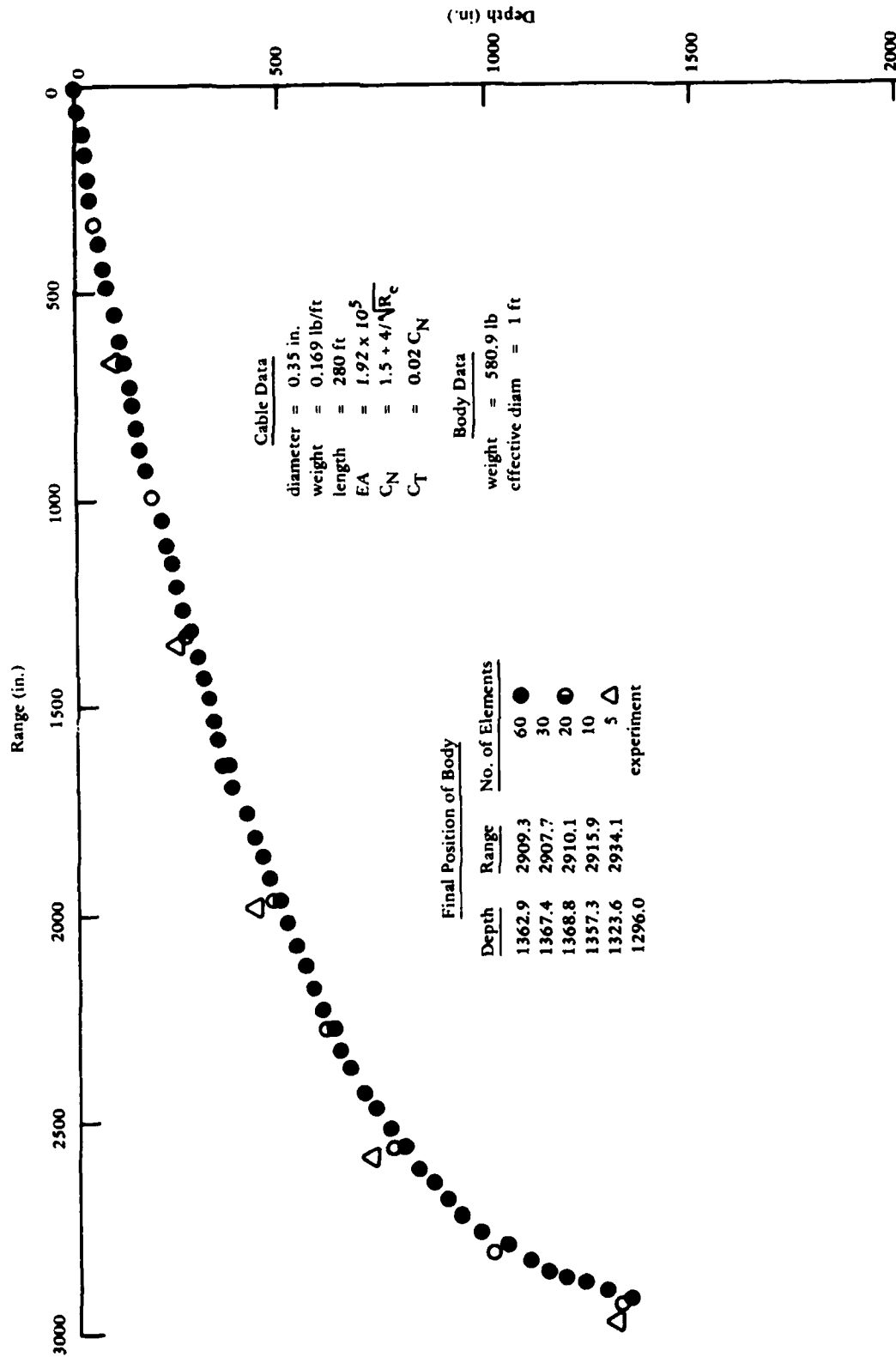


Figure 8-1. Towed body test case.

Table 8-4. USDRG SUBROUTINE

```

SUBROUTINE USDRG (IBOD,IDR,REN,RET,CN,CT,DRGDAT)
DIMENSION DRGDAT(20)

C
C   SPHERICAL BUOY
IF(IBOD.EQ.1) CN=.47
C   CYLINDRICAL BUOY OR CABLE
IF(IBOD.EQ.2.OR.IBOD.EQ.3) THEN
    CN=13.
    CT=10.
    IF(REN.LT.0.1) RETURN
    CN=1.5 + 4./SQRT(REN)
    CT=0.02*CN
ENDIF
RETURN
END

```

The steady-state towing configuration can be used as the starting point for a dynamic analysis in which the tow velocity and direction are changed. Table 8-5 shows the additional input required to continue from any of the previous converged static states to a dynamic solution where the tow point is moved. The key here is the change in flow-field velocity to zero and assigning the initial towing velocity to all of the nodes. The movement function given defines a sudden change of direction of the towing at the same velocity. Arbitrary movement can be defined.

Table 8-5. TOWED BODY DYNAMIC INPUT

Add this to the PROB data block:

TFUN

1,3,1,1,1,1,.866,90,.866

2,3,0,1,0,1,.5,90,.5

Add this to the SAO data block just prior to END

DYN

INIT,W5,212.8

MOVE,11,2,1,212.8,W9,2,2,212.8

TIME,,30 *FIND OWN TIME STEP

OUTP,W3,1

One general comment is needed on this problem. Care must be taken to assure all data are provided in consistent units. The choice of units is arbitrary since SEADYN does not presume any units (except for certain identified defaults). The choice made here is:

Length - inches

Force - pounds

Time - seconds

Note that lengths, diameters, forces, velocities, fluid properties, and gravitational acceleration all use these units. In all of the approaches, the tow cable drag coefficients are given by the user defined drag function shown in Table 8-4. The SEADYN default drag was on the body.

8.2 Buoy Relaxation Example

Consider a buoy restrained by a single line that is anchored to the seafloor. Assume that the buoy is snagged and hauled over to a certain position and held there. Then, after some time, the buoy and line are released to move dynamically to the original gravity loaded state. A SEADYN analysis of this scenario is presented in three stages:

- STAGE 1 - Establish the gravity-loaded initial state.
- STAGE 2 - Move the buoy over to the restrained (snagged) position.
- STAGE 3 - Release the restraint and follow the dynamics back to the initial state.

The actual execution of these stages is dependent on how the snagged position is defined. The problem statement implies the dynamics of the snagging operation are not important and only the calculation of the final static state in the restrained position is needed. Two different definitions of the snagged state are pursued here. The first presumes the snagged position of the buoy is known. The second presumes the magnitude and direction of the snagging load is specified.

Recalling the previous example problem, there are various approaches that can be used to get the initial static state. In both cases a gravity-load vertical state is used as a starting point. Loads or displacements are then applied, and iterations are carried to convergence at the final state. The dynamic phase is initiated simply by releasing the displacement constraint or force. Input for these examples is presented in Table 8-6. The second case is assumed as a restart from the first to demonstrate the RESTART section. The viscous relaxation method was chosen for both of these since it is most likely to remain stable and obtain convergence.

The dynamic solution used the direct iteration method since this has proven to be the most reliable. The time step was internally calculated since no external loading with its own time characteristics was present.

Table 8-6. BUOY RELAXATION INPUT

```

BUOY RELAXATION DEMONSTRATION PROBLEM  INITIAL POSITION GIVEN;

PROB;11,10,-2,1
FLUI;0,1
BODY;1, ,-.129,.16667
BLOC;1,1
MATE;1, ,.0125,.00055,W9,2.36,1
NODE
1, ,5.551
11,1,0,.104,W6,1,1,1
ELEM
1,1,2, ,1          *ISIGO=0 WITH NO LOADS SAME AS ISIGO=1
10,10,11, ,1
DEAD                *GET VERTICAL STATIC STATE
    SOLU, ,.01      *SET NUMERICAL DAMPING
    SAVE,1          *SAVE DEAD RESULTS
LIVE                *MOVE TO SNAGGED POSITION
    SOLU,VRR
    MOVE,1,1,1,4,1,1,-2
DYN                  *RELEASE FOR RELAXATION
    TIME, ,10
    OUTP, ,1
NEW
BUOY RELAXATION DEMONSTRATION PROBLEM  FORCES SPECIFIED;
REST;NEW,1          *GET THE LAST DATA SAVED FROM PREVIOUS DEAD
LIVE                *APPLY SNAG LOAD
    SOLU,VRR
    LOAD,1,1,W7,.115,.0265  *LVAR DEFAULTS
DYN                  *RELEASE FOR RELAXATION
    TIME, ,10
    OUTP, ,1
END

```

8.3 Anchor Last Example

An anchor last deployment scheme for tethered buoys is readily modeled in SEADYN. The starting state before the release of the anchor requires some careful consideration, however, when special situations are modeled. Static analysis problems can occur when floating lines with low initial tensions are involved. This presents a problem because slight changes in tension distribution can cause large position changes. In such situations it is best to make a reasonable guess on the initial state with no tensions and go immediately to the dynamic analysis following the release of the anchor.

This is a relatively simple problem intended to demonstrate the mechanics of getting the static and dynamic solutions accomplished. The problem presented here is sufficiently well-conditioned to allow the initial static state to be calculated. The problem characteristics are given in Figure 8-2. The input data are given in Table 8-7. The input assumes the anchor is released at time zero, and the top buoy is held for 10 seconds before it is released. The line between the two buoys is assumed to be neutrally buoyant. The initial state holds node 1 fixed and applies 1,000 pounds horizontal force to node 10. This stretches the neutrally buoyant line out straight and produces a catenary configuration in the heavy line. The guessed initial input uses unstretched lengths ($ISIGO = 1$) and the catenary line generator. The initial distance between nodes 4 and 10 is selected to give the desired line length for the catenary.

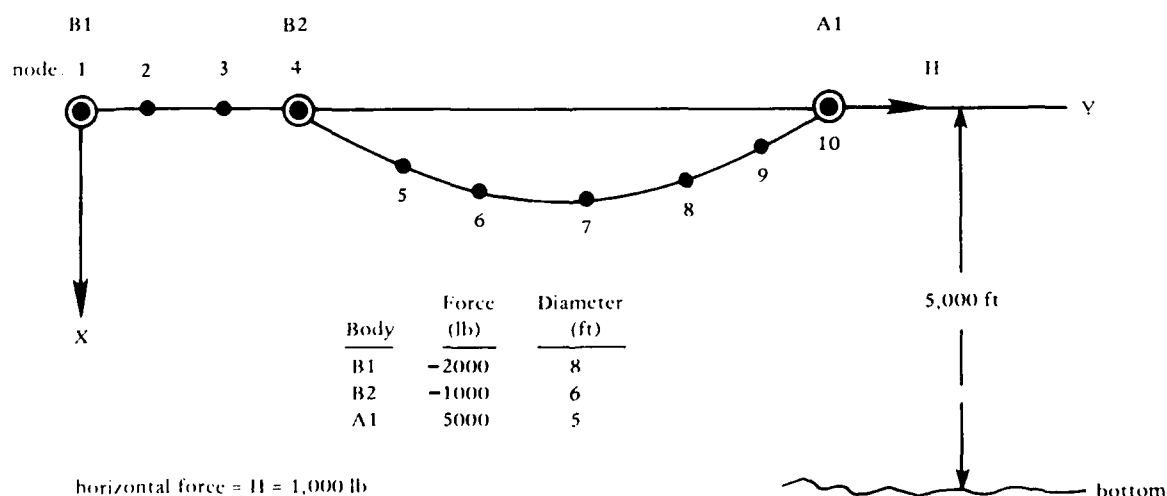


Figure 8-2. Anchor last model.

Table 8-7. ANCHOR LAST DEPLOYMENT INPUT

DOUBLE BUOY DEPLOYMENT USING ANCHOR LAST METHOD;

PROB;10,9,1,1

FLUI;0,1

LIMI

1,,1

*SURFACE

2,5000,1,,1

*BOTTOM

ILOC

1,1,4,3

*SURFACE LIMIT PLACED ON NODES 1 AND 4

2,10

*BOTTOM LIMIT PLACED ON NODE 10

BLOC

1,1

*BUOY 1 AT NODE 1

2,4

*BUOY 2 AT NODE 4

3,10

*ANCHOR AT NODE 10

NODE

1,W6,3,3,3

*HOLD TOP BUOY

4,,,1200,,3

*VERTICAL HOLD ON BUOY

10,,,2400,,3,,1

*HOLD ANCHOR, LEAVE HORIZONTAL FREE

NGEN;5,4,10,W7,2,.5,1000

*GENERATE CATENARY AND NODES 2,3

2,1,4

ELEM

1,1,2,,1,W8,1,1000

4,4,5,,2

9,9,10,,2

MATE

1,,.3,W9,1.E4,1

2,,.2,.5,W9,1.E4,1

BODY

1,, -2000,8,W8,20

*BUOY 1 WITH SURFACE DRAG

2,, -1000,6,W8,15

*BUOY 2 WITH SURFACE DRAG

3,,5000,5

*ANCHOR

DEAD

*SINGLE STEP VRR SOLUTION

SOLU,VRR

LOAD,1,10,W8,1000

I.VARY,1

DYN

*DROP ANCHOR

FIX,1,41

*CHANGE CODE TO ALLOW BUOY PULL-UNDER

FREE,101

TIME,,10.

OUTP,,2

DYN

*RELEASE BUOY 1

FREE,11,12

TIME,,1000

OUTP,,100

END

8.4 Payout From a Moving Ship Example

The deployment of a cable system by paying out line from a moving ship is demonstrated in this example. The problem selected is the final stages of deployment of a trapezoidal array system (Ref 15). Figure 8-3 shows the starting configuration. The system contains two clump anchors and two suspension buoys. In the sequence modeled, the first clump anchor is in place and the final anchor is being lowered by a line paying out from a cable ship. The payout rate and ship velocity are chosen so as to place the anchor at the desired point. The demonstration problem uses a straight line ship velocity (away from the cable system) of 1 ft/sec and a constant payout rate of 2 ft/sec.

The element model used is the simplest possible: only one element per line. The payout line element is assumed to grow to a maximum length of 10,000 foot and then be divided into two elements (mitosis). Up to this point element 5 and node 6 are not involved in the solution. When mitosis occurs, node 5 is placed at the dividing point, node 6 takes the position of the moving payout point, and element 5 enters the system as the upper half of the payout line with an unstretched length equal to the mitosis length. Element 5 is then the payout element, and element 4 maintains a constant unstretched length equal to the unstretched length just before mitosis minus the mitosis length. When the anchor (node 4) reaches the bottom, it is held fixed in three components.

Table 8-8 lists the input for a DYN analysis using the DIM method. Table 8-9 presents input for a TSSS analysis of the same time sequence. The VRR form was used, but the MNR method could also have been selected.

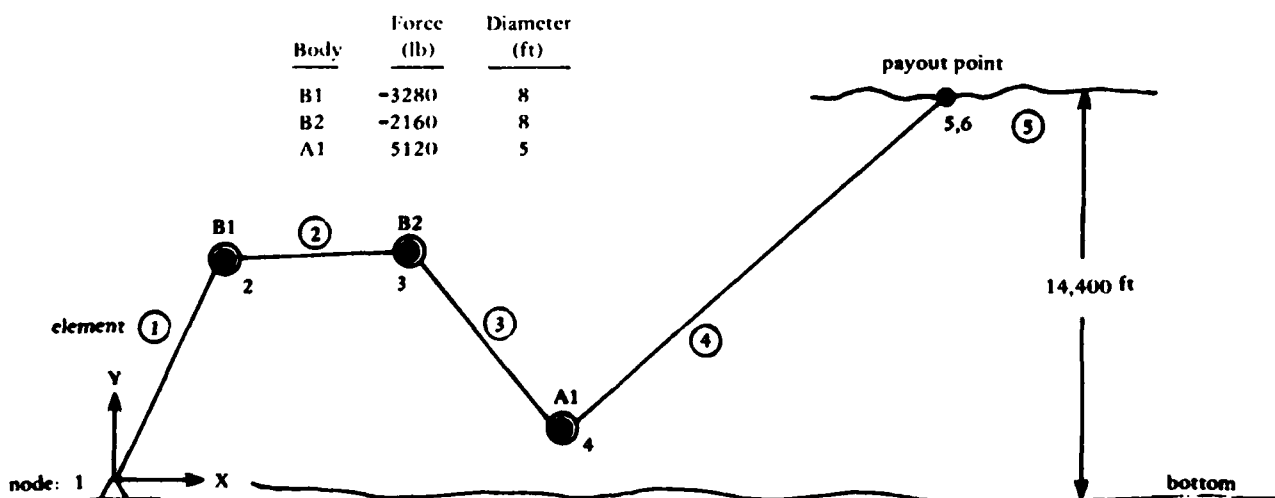


Figure 8-3. Payout model.

Table 8-8. PAYOUT FROM MOVING SHIP INPUT

PAYOUT FROM A MOVING SHIP;

```

PROB;6,5,-2,1
FLUI;14400,1
BODY
1,, -3280,8
2,, -2160,8
3,, +5120,5
BLOC
1,2
2,3
3,4          *CLUMP ANCHOR
LIMI;1,0,5,,1
LLOC;1,4      *STOP ANCHOR AT BOTTOM
MATE;1,,.1667,W9,1.E6,1
NODE
1,W6,3,3,3    *GLOBALLY FIX ANCHOR NODE
2,,7500,13300
3,,10400,14200
4,,23000,5700
5,,29739,14400,W6,2,2,2 *ORIGINAL PAYOUT NODE
6,,29739,14400,W6,1,1,1 *NEW PAYOUT NODE
ELEM
1,1,2,,1
5,5,6,,1
PAYO;1,5,4,5000,1,1,1  *DEFINE PAYOUT TOPOLOGY
DEAD
      SOLU,VRR
DYN
      MOVE,5,2,,1,1,W9,1
      PAYO,1,,2
      SOLU,W12,1          *RESTRICT TIME STEP GROWTH
      TIME,.5,4500
      OUTP,,100
END

```

Table 8-9. TSSS INPUT FOR PAYOUT PROBLEM

```

TSSS
      SOLU,VRR
      TIME,100,4000
      MOVE,5,2,,1,1,W9,1
      PAYO,1,,2
      OUTP,,100
END

```

8.5 Vessel Mooring Example

This example demonstrates input for a vessel mooring static analysis and the use of catenary elements. Two input examples are given: Table 8-10 uses all simplex elements to model the mooring, Table 8-11 uses both simplex and catenary elements.

The problem has a four-point mooring where each leg has a surface buoy and a hawser. The vessel is described by one master node and four slave nodes (at each hawser attachment point). The vessel dimensions are described in the SHIP record. If wind or surface current loads are to be applied to the vessel, the ship load data file must be included in the input (see Appendix A for format). Both example inputs have a ship load data file.

The second example has input for catenary elements for bottom interaction, such as laying down or picking up cable. Each leg has a catenary element for the first element by the anchor. The rest of the mooring leg is modeled with simplex elements to allow subsurface current loading or non-planar configurations. Catenary elements are constrained to lie in a plane.

Table 8-10. SIMPLEX ELEMENT INPUT FOR VESSEL MOORING PROBLEM

```

*THIS IS A MOORING SIMULATION - LIVE SOLUTION.  THE MOORING LEGS ARE
*1.5 INCH DIELOCK CHAIN, THE HAWSERS ARE 10 INCH CIRCUM POLYESTER.
*THERE ARE 15 ELEMENTS PER LEG (10 IN CHAIN AND 5 IN SYNTHETIC)  AND 1
*ELEMENT PER HAWSER WITH A + 6 KLB MOORING BUOY AT THE JUNCTION OF THE
*HAWSER AND MOORING LEG.
**
*MOORING LINE CATENARY CALCULATED USING A HORIZONTAL PRETENSION OF
*10,000 LBS, 9 SHOTS
*OF CHAIN, 1000-FT DEPTH TO FIND SPAN LENGTH AND ANCHOR LOCATION.
**
**
VESSEL MOORING...SYNTHETIC MOORING LEGS...10000 LB PRETENSION...9 SHOTS$
PROB
69,64,-3,1,,1
FLUID
,1
,2
BODY
1,, -6000,5W8,32
BLOC
1,61,64,1
NODE
1,, -2482,-3431,-1000,3,3,3
2,, -2482,3431,-1000,3,3,3
3,,2482,3431,-1000,3,3,3
4,,2482,-3431,-1000,3,3,3
21,, -2250,-3100,-1000
22,, -2250,3100,-1000
23,,2250,3100,-1000
24,,2250,-3100,-1000
41,, -2017,-2767,-985
42,, -2017,2767,-985
43,,2017,2767,-985
44,,2017,-2767,-985
61,, -397,-453
62,, -397,453
63,,397,453
64,,397,-453
65
66,, -110,-43,12,-65,-65,-65
67,, -110,43,12,-65,-65,-65
68,,110,43,12,-65,-65,-65
69,,110,-43,12,-65,-65,-65
NGEN
4,1,21,4,,,0,,,2
4,2,22,4,,,0,,,2
4,3,23,4,,,0,,,2
4,4,24,4,,,0,,,2
4,21,41,4,,,1,19.7,10000,2
4,22,42,4,,,1,19.7,10000,2

```

* BUOYS - 5 FT DIA, 6K BUOYANCY

* ANCHOR POSITS CALCULATED WITH 10K HORIZ
 * TENS USING 9 SHOTS/LEG OF 1-1/2 IN
 * CHAIN FROM ANCHOR TO BUOY - NOT RECALC
 * FOR 2 IN CHAIN

* BUOY NODE hawser length is 500 ft
 * BUOY NODE for 10k pretension
 * BUOY NODE
 * BUOY NODE
 * SHIP NODE
 * SLAVE NODES TO SHIP,
 * HAWSER ATTACHMENT POINTS

Table 8-10. (CONT)

4,23,43,4,,,1,19.7,10000,2
 4,24,44,4,,,1,19.7,10000,2
 4,41,61,4,,,2,19.7,10000
 4,42,62,4,,,2,19.7,10000
 4,43,63,4,,,2,19.7,10000
 4,44,64,4,,,2,19.7,10000
 ELEM
 1,1,5,,1 * LINE 1
 10,37,41,,1,,4,1
 11,41,45,,1W9,10800
 15,57,61,,1,,4,1,10800
 16,61,66,,2
 17,2,6,,1 * LINE 2
 26,38,42,,1,,4,1
 27,42,46,,1W9,10800
 31,58,62,,1,,4,1,10800
 32,62,67,,2
 33,3,7,,1 * LINE 3
 42,39,43,,1,,4,1
 43,43,47,,1W9,10800
 47,59,63,,1,,4,1,10800
 48,63,68,,2
 49,4,8,,1 * LINE 4
 58,40,44,,1,,4,1
 59,44,48,,1W9,10800
 63,60,64,,1,,4,1,10800
 64,64,69,,2
 MATE
 1,,.25,19.7W9,34380000,1 * 1.5 INCH CHAIN
 2,,.26,.945W9,17200000,1 * 3.25 INCH DIA POLYESTER DOUBLE BRAID
 LIMIT
 1,,1
 2,-1000,.1,,1
 LLOC
 1,61,64,1
 2,5,44,1
 SHIP
 65,1,,420,,,251,4500,9300,240,86,24,173250,186,W20,1.E22,1.E22,W23,1000
 TENS
 1,10,1,1
 17,26,1,2
 33,42,1,3
 49,58,1,4
 16,,,10000
 32,,,10000
 48,,,10000
 64,,,10000
 (
 SHIP LOAD FILE FOR USS PIGEON(ASR-21);WIND-STATMOOR,CURR-DESTROYER TESTS

Table 8-10. (CONT)

| POUNDS | POUNDS | FEET | FT/SEC | | | | | |
|------------|-----------|------------|----------|----------|----------|-----------|---------|--|
| 2.60E+02 | 1.5E+03 | 3.9E+03 | 2.6E+02 | 4.0E+01 | 0.8E+01 | 0.832E+05 | | |
| 1 17 | 1.E+00 | 1.014E+02 | | | | | | |
| 0.E+01 | 1.E+01 | 2.25E+01 | 3.5E+01 | 4.5E+01 | 5.5E+01 | 6.75E+01 | 8.E+01 | |
| 9.E+01 | 1.E+02 | 1.125E+02 | 1.25E+02 | 1.35E+02 | 1.45E+02 | 1.575E+02 | 1.7E+02 | |
| 1.8E+02 | | | | | | | | |
| 3.857E+04 | 0.E+01 | 0.E+01 | | | | | | |
| 3.917E+04 | 1.164E+04 | -5.215E+05 | | | | | | |
| 4.080E+04 | 2.888E+04 | -1.078E+06 | | | | | | |
| 4.106E+04 | 4.882E+04 | -1.434E+06 | | | | | | |
| 3.868E+04 | 6.373E+04 | -1.530E+06 | | | | | | |
| 3.355E+04 | 7.474E+04 | -1.444E+06 | | | | | | |
| 2.429E+04 | 8.132E+04 | -1.096E+06 | | | | | | |
| 1.450E+04 | 8.226E+04 | -5.443E+05 | | | | | | |
| 8.070E+03 | 8.202E+04 | -2.341E+04 | | | | | | |
| 3.468E+03 | 8.223E+04 | 5.625E+05 | | | | | | |
| -1.113E+03 | 8.152E+04 | 1.196E+06 | | | | | | |
| -1.079E+04 | 7.541E+04 | 1.608E+06 | | | | | | |
| -2.383E+04 | 6.466E+04 | 1.721E+06 | | | | | | |
| -3.770E+04 | 4.969E+04 | 1.625E+06 | | | | | | |
| -4.681E+04 | 2.938E+04 | 1.226E+06 | | | | | | |
| -4.564E+04 | 1.180E+04 | 5.939E+05 | | | | | | |
| -4.408E+04 | 0.000E+01 | 0.000E+01 | | | | | | |
| 1 15 | 1.0E+03 | 0.167E+01 | | | | | | |
| 0.E+01 | 1.5E+01 | 3.0E+01 | 4.5E+01 | 6.0E+01 | 7.0E+01 | 8.0E+01 | 9.0E+01 | |
| 1.0E+02 | 1.1E+02 | 1.2E+02 | 1.35E+02 | 1.5E+02 | 1.65E+02 | 1.8E+02 | | |
| 1.40E+03 | 0.000E+01 | 0.000E+01 | | | | | | |
| 1.37E+03 | 5.400E+03 | -9.640E+04 | | | | | | |
| 1.30E+03 | 1.080E+04 | -9.500E+04 | | | | | | |
| 1.20E+03 | 1.584E+04 | -7.560E+04 | | | | | | |
| 1.08E+03 | 1.920E+04 | -6.000E+03 | | | | | | |
| 9.00E+02 | 2.040E+04 | 6.000E+04 | | | | | | |
| 6.00E+02 | 2.112E+04 | 1.440E+05 | | | | | | |
| 0.00E+01 | 2.184E+04 | 2.160E+05 | | | | | | |
| -8.00E+02 | 2.208E+04 | 2.700E+05 | | | | | | |
| -1.92E+03 | 2.208E+04 | 3.240E+05 | | | | | | |
| -2.12E+03 | 2.160E+04 | 3.840E+05 | | | | | | |
| -3.16E+03 | 2.088E+04 | 4.100E+05 | | | | | | |
| -3.44E+03 | 1.704E+04 | 4.080E+05 | | | | | | |
| -3.56E+03 | 1.056E+04 | 2.925E+05 | | | | | | |
| -3.60E+03 | 0.000E+01 | 0.000E+01 | | | | | | |

)

DEAD

FIX, 3, 651, 652, 653, 654, 655, 656

LIVE

WIND, 27, 0, 1

SURF, 1.69, 0, 1

FREE, 651

STEP, 10, 10

SOLU, VRR, .01, .01

END

* FREE VESSEL TO MOVE IN X DIRECTION

**Table 8-11. CATENARY AND SIMPLEX ELEMENT INPUT
FOR VESSEL MOORING PROBLEM**

```

VESSEL 4 POINT MOORING WITH 10KLB PRETENSION $$
PROB
29,24,-3, 1,W6,1
FLUID
,1
,2
NODE
1,,-3218.5,-4596.5,-1000,3,3,3      * LEG 1 ANCHOR
2,,-3218.5,4596.5,-1000,3,3,3      * LEG 2 ANCHOR
3,,3218.5,4596.5,-1000,3,3,3      * LEG 3 ANCHOR
4,,3218.5,-4596.5,-1000,3,3,3      * LEG 4 ANCHOR
*
* NOTE, KSKP GENERATES NODES IN A STRAIGHT LINE TO BUOY.
*
5,,-2805.5,-4006.6,-999.9          * LEG 1 CHAIN/SYN
17,4,-288.7,-412.3,0              * BUOY/HAWSER
6,,-2805.5,4006.6,-999.9          * LEG 2 CHAIN/SYN
18,4,-288.7,412.3,0              *
7,,2805.5,4006.6,-999.9          * LEG 3 CHAIN/SYN
19,4,288.7,412.3,0              *
8,,2805.5,-4006.6,-999.9          * LEG 4 CHAIN/SYN
20,4,288.7,-412.3,0              *
25                                  * SHIP CENTER OF GRAVITY
26,,-110,-43,12,-25,-25,-25      * HAWSER/SHIP, LEG 1
27,,-110,43,12,-25,-25,-25       * HAWSER/SHIP, LEG 2
28,,110,43,12,-25,-25,-25       * HAWSER/SHIP, LEG 3
29,,110,-43,12,-25,-25,-25      * HAWSER/SHIP, LEG 4
NGEN
0,4,8,0,0,0,3,23,10000          * CATENARY REFERENCED BY ELEM
0,3,7,0,0,0,3,23,10000
0,2,6,0,0,0,3,23,10000
0,1,5,0,0,0,3,23,10000
1,17,26,4,,21                   * GENERATE HAWSER NODES
1,18,27,4,,22
1,19,28,4,,23
1,20,29,4,,24
ELEM
1,1,5,,1,-1W10,-4               * LEG 1 CATENARY
2,5,9,,2W9,10900
6,21,26,,2,,4W9,10900
7,2,6,,1,-1W10,-3               * LEG 2 CATENARY
8,6,10,,2W9,10900
12,22,27,,2,,4W9,10900
13,3,7,,1,-1W10,-2              * LEG 3 CATENARY
14,7,11,,2W9,10900
18,23,28,,2,,4W9,10900
19,4,8,,1,-1W10,-1              * LEG 4 CATENARY
20,8,12,,2W9,10900
24,24,29,,2,,4W9,10900

```


Table 8-11. (CONT)

TENS

2,4,1,,10900
8,10,1,,10900
14,16,1,,10900
20,22,1,,10900
5,6,,,10000
11,12,,,10000
17,18,,,10000
23,24,,,10000

* MOORING LEGS

* HAWSERS

BODY

1,, -6000,5W8,32

BLOC

17,20,1,1

LIMIT

1,0
2,-1000,,,1

LLOC

1,17,20,1
2,2,12,1

MATE

1,,.75,23W9,105.6E+06,1
2,,.26,.919W9,.153E+07,1

* 1 1/2 IN DILOK

* 10" DOUBLE BRAID POLYESTER

SHIP

25,1W7,251,4500,9300,240,86,24,173250,186W20,1.E5,1.E5,1.E3,1000

(

SHIP LOAD FILE FOR USS PIGEON(ASR-21);WIND-STATMOOR,CURR-DESTROYER TESTS

POUNDS POUNDS FEET FT/SEC

| | | | | | | | | |
|------------|-----------|-----------|----------|----------|----------|-----------|---------|--|
| 2.60E+02 | 1.5E+03 | 3.9E+03 | 2.6E+02 | 4.0E+01 | 0.8E+01 | 0.832E+05 | | |
| 1 17 | 1.E+00 | 1.014E+02 | | | | | | |
| 0.E+01 | 1.E+01 | 2.25E+01 | 3.5E+01 | 4.5E+01 | 5.5E+01 | 6.75E+01 | 8.E+01 | |
| 9.E+01 | 1.E+02 | 1.125E+02 | 1.25E+02 | 1.35E+02 | 1.45E+02 | 1.575E+02 | 1.7E+02 | |
| 1.8E+02 | | | | | | | | |
| 3.857E+04 | 0.E+01 | 0.E+01 | | | | | | |
| 3.917E+04 | 1.164E+04 | 5.215E+05 | | | | | | |
| 4.080E+04 | 2.888E+04 | 1.078E+06 | | | | | | |
| 4.106E+04 | 4.882E+04 | 1.434E+06 | | | | | | |
| 3.868E+04 | 6.373E+04 | 1.530E+06 | | | | | | |
| 3.355E+04 | 7.474E+04 | 1.444E+06 | | | | | | |
| 2.429E+04 | 8.132E+04 | 1.096E+06 | | | | | | |
| 1.450E+04 | 8.226E+04 | 5.443E+05 | | | | | | |
| 8.070E+03 | 8.202E+04 | 2.341E+04 | | | | | | |
| 3.468E+03 | 8.223E+04 | 5.625E+05 | | | | | | |
| -1.113E+03 | 8.152E+04 | 1.196E+06 | | | | | | |
| -1.079E+04 | 7.541E+04 | 1.608E+06 | | | | | | |
| -2.383E+04 | 6.466E+04 | 1.621E+06 | | | | | | |
| -3.770E+04 | 4.969E+04 | 1.625E+06 | | | | | | |
| -4.681E+04 | 2.938E+04 | 1.226E+06 | | | | | | |
| -4.564E+04 | 1.180E+04 | 5.939E+05 | | | | | | |
| -4.408E+04 | 0.000E+01 | 0.000E+01 | | | | | | |
| 1 15 | 1.0E+03 | 0.167E+01 | | | | | | |
| 0.E+01 | 1.5E+01 | 3.0E+01 | 4.5E+01 | 6.0E+01 | 7.0E+01 | 8.0E+01 | 9.0E+01 | |
| 1.0E+02 | 1.1E+02 | 1.2E+02 | 1.35E+02 | 1.5E+02 | 1.65E+02 | 1.8E+02 | | |

Table 8-11. (CONT)

1.40E+03 0.000E+01 0.000E+01
 1.37E+03 5.400E+03-9.640E+04
 1.30E+03 1.080E+04-9.500E+04
 1.20E+03 1.584E+04-7.560E+04
 1.08E+03 1.920E+04-6.000E+03
 9.00E+02 2.040E+04 6.000E+04
 6.00E+02 2.112E+04 1.440E+05
 0.00E+01 2.184E+04 2.160E+05
 -8.00E+02 2.208E+04 2.700E+05
 -1.92E+03 2.208E+04 3.240E+05
 -2.12E+03 2.160E+04 3.840E+05
 -3.16E+03 2.088E+04 4.100E+05
 -3.44E+03 1.704E+04 4.080E+05
 -3.56E+03 1.056E+04 2.925E+05
 -3.60E+03 0.000E+01 0.000E+01

)

DEAD

OUTP,W4,1

FIX,3,255,256

END

9.0 SUMMARY OF MAJOR DIAGNOSTIC MESSAGES

The SEADYN program makes some checks of the input data and attempts to aid the user in finding errors by printing various messages. No attempt has been made to be comprehensive in this feature since it is very difficult to foresee and/or detect many of the possible errors. The input routines that process the PROB and SAO data sets produce various diagnostics that evaluate errors detected in the input. These messages are generally self-explanatory. They deal mainly with program restrictions, such as the maximum number of items allowed, or the completeness and consistency of the data provided. The user should have little difficulty interpreting the problem detected, and should be able to make appropriate corrections with the aid of this manual.

During the processing of the SAO options, checks are made on the validity of the requests and the convergence of the analysis procedures. The messages that can be printed are listed below with a brief description of the probable cause and/or cure. The action taken after the error detection is indicated by the following codes:

- (F) Fatal, run aborted.
- (N) Abort analysis case and seek a new problem definition by searching the deck for a NEW record.
- (O) Abort present SAO activity and go to the next SAO data set.
- (S) Skip this request and go to the next card.
- (C) Continue calculation with action as indicated.

Note that the diagnostic messages listed in each section are in alphabetical order of the first word.

9.1 System Description Checks

ROUTINE

- (F) BASE (IBASE) COMMON ON TAPE LARGER THAN SPACE AVAILABLE

RESTRT

The data saved on the file requires more storage than is presently available. The two numbers printed are the required and current values of NCOM (NICOM).

- (F) DATA PRECISION ON TAPE DOES NOT AGREE WITH THE VERSION OF THE PROGRAM

RESTRT

The tape was written by an incompatible version of the program.

- (F) DEPTH CORRECTION ERROR -- SHIP DRAFT EXCEEDS WATER DEPTH

DEPCOR

Check SHIP input Words 12 and 23

- (N) ELEMENT GENERATION ERROR ON LINE XX

INPELT

First element was not input, or element cards are not in increasing element number order, or last element was not input.

- (F) ERROR IN (WIND/CURRENT) LOAD TABLE ON SHIP XXX
HEADING = XXX
LAST TABLE ENTRY = XXX
SYMMETRY FLAG = XXX

LDNTRP

Heading requested exceeds the largest value in the table. Check ship load input table.

- (N) IMPROPERLY DEFINED MOORING BUOY AT NODE XXX NO. OF SLAVES FOUND = XXX

CONCNT

Moorings require at least two slaves.

- (F) INSUFFICIENT SLAVES TO DEFINE A RIGID BODY AT NODE XXX NO. OF SLAVES FOUND XXX

CONCNT

Towed rigid bodies (not mooring buoys) require at least one attachment slave plus one slave for the center of gravity and one for the center of buoyancy.

ROUTINE

(N) NODES OMITTED IN GENERATION

INPT

Message is followed by a list of ones and zeros (ten per line) corresponding to the nodes in the system. The zeros indicate which nodes were not defined either by input or generation sequence. All nodes must be accounted for. Following the integers, the nodal coordinates are printed to aid in finding the error. The problem usually comes from improper node generation input.

(N) SHIP DATA INPT ERROR ON XXX (Ship No.)
NO. OF SHIPS ON FILE = XXX
LOAD FUNCTION OPTION = XXX

SHPDEF

Blank record for moored ship data requested
definition of ship from load file with no
load file defined.

OR

Attempt ship scaling with no load file defined.

(F) TAPE LABEL XXXXXX DOES NOT AGREE WITH XXXXXX

RESTR

The label check failed on restart. The first six characters on the RESTART title record did not agree with the check word given.

9.2 Subanalysis Option Errors

ROUTINE

- (S) ANCHOR NOT ON BOTTOM AT NODE XXX
SKIP REQUEST FOR XXXX COMCHK

Anchor weight is not sufficient to hold the lines at this node and it has been lifted off the bottom or has not reached the bottom.

- (S) CAPACITY AND COMPONENT ID BOTH ZERO
SKIP REQUEST FOR XXXX (CTYPE) COMCHK

Check input card.

- (F) INCONSISTENT ANALYSIS REQUEST ON RESTART AT
LINE XX SEADYN

Attempted a restart of DEAD, LIVE, or DYN, and the file was not from that type of subanalysis.

- (N) INSUFFICIENT STORAGE TO PROCEED
COMMON SIZE = XXX
STORAGE NEEDED = XXX
HALF BANDWIDTH = XXX
DEGREES OF FREEDOM = XXX
BASE SIZE = XXX MANIPR

Subanalysis request (DEAD, LIVE, DYN) cannot be processed due to storage limitations. Problem must be reformulated or NCOM increased. (See Appendix I)

- (S) INVALID COMPONENT TYPE
SKIP REQUEST FOR XXXX COMCHK

The component number is not one recognized by the component inventory.

- (S) MORE THAN TWENTY LINES ON ANCHOR AT NODE XXX
SKIP REQUEST FOR XXXX COMCHK

The fixed node where anchor capacity check requested has too many element connected to it to make the check.

- (S) NO LINES CONNECTED TO NODE XXX
SKIP REQUEST FOR XXXX COMCHK

Check node number.

ROUTINE

- (O) NOT ENOUGH STORAGE FOR MODE SHAPES
NEED XXXX, WITH XXXX AVAILABLE

MANIPR

Storage inadequate for MODE analysis.
Increase NCOM. (See Appendix I.)

- (N) NOT ENOUGH STORAGE FOR STRUM MODES SHAPES
NEED--XXX WITH XXX AVAILABLE ON STRING XX

Storage inadequate for strum processing.
Increase NCOM.

- (C) SLOW CONVERGENCE ON STEP XX
LAST FOUR VELOCITY NORMS XXXX XXXX XXXX XXXX
LAST RESIDUAL NORMS XXXX XXXX XXXX

VESREL

This is a progress report on the VRR
iterations. Damping is reduced or step size
is increased depending on the pattern of
residual norm changes. Repeated occurrence
of these messages is a signal that the
solution is in difficulty. Consider
modifying initial damping or the model.

- (N) UNRECOGNIZED ANALYSIS OPTION = XXXX
TRY TO GET TO NEXT CASE

Usually indicates improper numbers of
cards or cards out of sequence.

9.3 Solution Option Execution Messages

ROUTINE

- (F) AUTOSTEP SELECTION FAILED TO REDUCE RESIDUAL
ON STEP XXX RNORM = XXX CHECK LOADS AND
VARIATION CODES

NRAPIT

The load or displacement variation
codes may be inconsistent.

- (F) CATENARY BOTTOM ITERATION FAILED TO CONVERGE ON
STEP XX INCREMENT YY ELEMENT ZZ

CATFRC

Usually signals gross node input error
or divergent solution. This message
is followed by geometry and solution
data for the catenary in its local
coordinate system.

- (F) CATENARY GEOMETRY ERROR ON ELEMENT XX
DEGENERATE BASE VECTOR FOR LOCAL COORDINATES

CATFRC

Catenary element geometry error.
Check node and current input.

- (F) DID NOT GET LOCAL CATENARY PLANE ON ELEMENT
XX COMPONENTS OF THE ELEMENT VECTOR (FROM
1ST TO 2ND NODE)

CATFRC

Usually indicates gross node input
error or divergent solution. Will
occur in the unlikely event that
the two nodes on the catenary coincide.

- (N) DIVERGENCE ON STEP XXX AT XXXXX
(load factor or time)

NRAPIT

Signals abort of MNR method after KNVRT
successive step size reductions.

- (C) DIVERGING ITERATION INCREMENT, TIME,
STEP SIZE, KOUNT AA BB CC DD
LAST TWO NORMS XXXX XXXX

STPDYN

Signals a lack of convergence in the
Direct Numerical Integration time
domain analysis. The various sit-
uations that lead to this message
are:

ROUTINE

- (a) Acceleration more than doubled in one iteration on the component with the largest acceleration.
- (b) The displacement norm exceeds 1×10^{12} .
- (c) The displacement norm increased in three successive iterations.
- (d) The largest displacement increment exceeds a magnitude of 1×10^{10} .
- (e) The number of iterations exceeds LMITER on SOLU record. This will be followed by a reduction in time step subject to the limits of KNVRT on the SOLU record.

(C) DIVIDED STEP SIZES XXXX XXXX, INC, KOUNT XX XX STEP

A constraint overshoot was detected in the RFB incremental analysis. The step was divided into two parts as indicated by the message. The first part represents the portion of the original step used to get to the constraint. The remaining portion of the step was then taken with the constraint imposed. A full step size is used after successful completion of the divided step. Repeated divided steps with multiple constraints can cause the solution to fail. In that case small step sizes should be used.

(N) DYNAMIC SOLUTION DOES NOT CONVERGE AT TIME STPDYN
XXXX WITH A TIME INCREMENT OF XXX
LAST TWO NORMS XXXX XXXX

Signals abort of DIM solution after KNVRT successive step size reductions.

(N) FAILURE IN VISCOUS RELAXATION SOLUTION

All attempts to get convergent VRR iterations have failed. This occurs when repeated norm increases or large strains are encountered.

| | <u>ROUTINE</u> |
|--|----------------|
| (N) IMPROPERLY DEFINED DYNAMIC PROBLEM. NO MASS AT NODE XX DIRECTION YY (DEGREE OF FREEDOM ZZ) | EQTION |
| <p>A material property input error has been made or zero length element has been included in the model that is not one of those reserved for payout.</p> | |
| (F) INCONSISTENT LENGTHS, MUST ABORT ON ELEMENT EE XX YY ZZ | COMPRP |
| <p>The lengths assigned to the two sub-elements (YY and ZZ) in multimaterial payout do not add up to the total element length (XX). This may be a signal of numerical problems caused by large differences in EA.</p> | |
| (C) INCREASING RESIDUAL NORM ON STEP XX LAST THREE RESIDUAL NORMS XXXX XXXX XXXX | VISREL |
| <p>The VRR iterations have produced successive increases in the force residual norm. Divergence is indicated. Damping is increased and the iteration restarted.</p> | |
| (C) INCREASING VELOCITY NORM ON STEP XX NORM VALUES XXXX XXXX XXXX XXXX REPEAT STEP | VISREL |
| <p>Indicates a strongly increasing velocity behavior in the VR iterations. This is assumed to be a signal of divergence if it is occurring repeatedly. Damping is increased and the step is repeated. In some situations this message will occur many times without an abort. This is because it gets intermittent good results rather than successive increases. This usually means the heuristic scheme for adjusting the solution parameters does not work well for the problem. In some situations this can be avoided by selecting a larger damping at the start.</p> | |

ROUTINE

- (C) LAST TWO RESIDUAL AND DISPLACEMENT NORMS XXXX
XXXX XXXX XXXX KOUNT = XXXX

NRAPIT

The four values printed are the residual and displacement norms for iteration (i-1) and iteration i in the MNR method. This message signals an increasing norm indicative of divergence or a lack of convergence in the number of iterations given by KOUNT. This occurs when norm values are very large (greater than 10,000) or repeated increases are detected. This will be followed by a reduction of step size subject to the limits of KNVRT on the SOLU record.

- (C) MOMENT SIGN CHANGE ON SHIP (BUOY), DIRECTION
XX YY

NRNORM

When the MNR solution detects a change in the sign of the moment residual component on a body, that component is temporarily fixed at the approximate zero moment position. This fix is held until the iteration is converged, then the fix is released and the iteration is repeated.

- (F) MULTIMATERIAL PAYOUT ITERATION FAILED TO
CONVERGE FOR ELEMENT XX AND LOADS YY ZZ

COMPRP

Payout mitosis has reached a point where the next element to be paid out has a different material than the preceding one. The iteration to solve for the composite load/strain state has failed to balance the load in the two sub-elements. This may be a signal of numerical problems caused by large differences in EA.

- (C) NEW DAMPING = XXXX

NRAPIT
VISREL

Indicates the action taken.

- (F) NEW PAYOUT ELEMENT XX DOES NOT HAVE A COMMON
NODE WITH THE CURRENT ELEMENT

PAYUPD

Check the payout topology and element numbering.

ROUTINE

(C) NEW SHIP (BUOY) ANGLE DAMPING ON INC,
KOUNT, SHIP (BUOY)
AA BB CC DD
LAST TWO MOMENT INCS AND PRESENT ANGLE
EE FF GG

NRNORM

When the MNR or VRR solution detect large oscillation of body angle responses, the angle damping is increased.

AA is step number
BB is iteration number
CC is body identification number
DD is the new damping parameter
EE, FF are the values of the body moment residual changes for the last two iterations
GG is the response angle in radians

(C) NEW STEP SIZE = XXXX

NRAPIT
STPDYN
VISREL

Indicates the action taken.

(C) SINGULAR EQUATION WITH NUMERICAL DAMPING
FACTOR OF XXXX

NRAPIT

This message is printed from the MNR method. See the "SOLUTION FAILED" message for further explanation. This message is printed after the trials described in that note. This will be followed by a reduction in step size subject to the limits of KNVRT on the SOLU record.

(N,C) SOLUTION FAILED DUE TO A ZERO PIVOT ON
ROW XX NODE YY DIRECTION ZZ THE
RECIPROCAL OF THE PRECEDING PIVOTS ARE

SLVCBN

The simultaneous equations in the sub-routine SLVCBN (DEAD, LIVE, DYN) are singular or sufficiently ill-conditioned to appear singular. Check node YY for proper constraint. This also occurs in poorly tensioned (soft) initial configurations. Check for zero tensions, etc. It can also occur from wildly divergent VRR iterations due to improperly posed problem or poor choice of parameters (usually damping is too small).

ROUTINE

If this occurs with the RFB method or in the steady-state response calculations in FREQ SAO, the case is terminated.

If this occurs during VRR or MNR analysis, various attempts are made to remove the singularity and repeat the step. Failing in these, the case is terminated. The disparity in the size of the pivot terms is an indication of ill-conditioning.

(C,N) STEP SIZE REDUCED TO XXXX ON INCREMENT XX
RESIDUAL AND DISPLACEMENT NORMS XXXX XXXX

NRAPIT

This message follows the reduction of step size in the MNR method. It will follow either of the previous two messages when the number step size reductions is within the limits imposed by KONVRT on the SOLU record.

(C,N) TIME STEP REDUCED TO XXXX ON STEP XXXX
NIRM = XXXX

STPDYN

This message follows the reduction of step size in a DIM solution. It will follow the previous message when the number of step size reductions is within the limits imposed by KONVRT on the SOLU record.

(N) VISCOUS RELAXATION SOLUTION HAS FAILED TO
CONVERGE IN XX ITERATIONS

VISREL

Check solution parameters--particularly initial damping. The message is followed by a series of solution parameters. It requires intimate familiarity with algorithm to interpret.

(C) WARNING - MULTIMATERIAL, SUDDENLY REMOVED ON
ELEMENT XX AT TIME YY

PAYUPD

When the element being reeled in has a different material than the one just removed there will be a multi-material element until enough length is subtracted to remove the influence of the old element. When this happens in one iteration there will be a pulse on the element mass. This message reports that occurrence.

10.0 REFERENCES

1. Naval Civil engineering Laboratory. Contract Report CR-82.019: SEADYN Mathematical Manual, by R.L. Webster, Consulting Engineer. Brigham City, UT, Apr 1982 (Contract No. N62474-81-C-9391).
2. _____. Contract Report CR-82.018: SEADYN programmer's reference manual, by R.L. Webster, Consulting Engineer. Brigham City, UT, Apr 1982 (Contract No. N62474-81-C-9391).
3. _____. Contract Report CR-82.017: A compendium of cable properties, by J.F. Wadsworth, III. Oxnard, Calif., Western Instrument Corporation, Apr 1982 (Contract No. N68305-80-C-0004).
4. _____. Contract Report CR-82.015: Validation of computer models of cable system dynamics, by David B. Dillon. Rockville, MD, EG&G Washington Analytical Services Center Inc., Apr 1982 (Contract No. N68305-80-C-0020).
5. Naval Civil Engineering Laboratory. Technical Note 1634: STATMOOR - A Single-Point Mooring Static Analysis Program, by J.V. Cox, Port Hueneme, CA, June 1982.
6. P.A. Palo, D.J. Meggitt, and W.J. Nordell, "Dynamic Cable Analysis Models," 15th Annual Offshore Technology Conference, pp 487-494, 2-5 May 1983, Houston, TX.
7. R.L. Webster, P.A. Palo, "Experiences in the Development of a General-Purpose Mooring Analysis Capability," Ocean Structural Dynamics Symposium '86 Proceedings, Sept 9-11 1986, Oregon State University, Corvallis, OR, pp 421-435.
8. R.L. Webster. "An application of the finite element method to the determination of nonlinear static and dynamic responses of underwater cable structures," Ph.D. thesis, Cornell University. Ithica, NY, Jan 1976. (Also available as General Electric Co. Report TJS-R76EMH12.)
9. _____. "On the static analysis of structures with strong geometric nonlinearities," Computers and Structures Journal, vol 11, no. 1/2, 1980, pp 137-145.
10. R.D. Cook. Concepts and applications of finite element analysis. John Wiley & Sons, Inc., NY, 1974.
11. R.L. Webster and W.R. McCreight. "Analysis of deep sea moor and cable structures," Paper presented at the 11th Annual Offshore Technology Conf., Houston, TX, May 1979 (OTC paper 3623).
12. General Electric Co., Electronic System Div. Report DSSM-5: Final engineering report, deep sea moor computer summation program," by R.L. Webster. Syracuse, NY, (Contract N62477-76-C-0002) Revised Sep 1978.

13. USCG Research and Development Center. Report No. CG-D-39-80: The dynamic behavior of nylon and polyester line, by K.R. Bitting. Groton, CT, Apr 1980.
14. Naval Research Laboratory. Memorandum Report 3383: SEACON II strumming predictions, by R.A. Skop, O.M. Griffin, R.E. Ramberg. Washington, DC, Oct 1976.
15. Naval Underwater Systems Center. Report No. 4141: Dynamics of trapezoidal cable arrays, by G.T. Griffin and K.T. Patton. New London, CT, Mar 1972.
16. Naval Ship Research and Development Center. Report 3376: NSRDC ship-motion and sea-load computer program, by W.G. Meyers, D.J. Sheridan, and N. Salvesen. Bethesda, MD, Feb 1975.
17. Naval Facilities Engineering Command. DM-26: Design manual: Harbor and coastal facilities, Washington, DC, Jul 1968.
18. G. Hughes. "The air resistance of ships' hulls with various types of distributions of superstructures," Institution of Engineers and Shipbuilders in Scotland, Transactions, vol 75, pt 5, Mar 1932, pp 302-339 and discussion pp 339-367.
19. H.E. Rossell and L.B. Chapman, editors. "Principles of Naval Architecture," Society of Naval Architects and Marine Engineers, New York, 1939.
20. Hydronautics, Inc. Technical Report 7096-1: Forces on ships moored in protected waters, by R. Altman, Jul 1971.
21. David Taylor Naval Ship Research and Development Center. Report 4610: Experimental determination of hydrodynamic loading for ten cable fairing models, by R. Folb. Bethesda, MD, Nov 1975.
22. R.L. Webster and P.A. Palo. "Experience in the development of a general purpose mooring analysis capability," Ocean Structural Dynamics Symposium, Corvallis, OR, 1986.

Appendix A

SHIP WIND AND CURRENT LOADS DATA FILE

The purpose of the ship load data file is to provide static loads for ships and other rigid bodies for wind and surface currents at various headings. The file can be constructed and saved as a library of static load functions; Appendix F describes the procedure used in DSSM to scale loads between similar ships using this library of files. Input is provided in a FORTRAN rigid format. Each ship load data set consists of:

- SHIP LOAD TITLE RECORD
- UNIT LABEL RECORD
- SHIP PARAMETERS
- WIND RECORD
- WIND HEADING RECORD(S)
- WIND FORCE RECORD(S)
- SURFACE CURRENT RECORD
- SURFACE CURRENT HEADING RECORD(S)
- SURFACE CURRENT FORCE RECORD(S)

Each of the data records is described below.

The free-field input routine will automatically read these rigid format records when they are placed between the () delimiter records (see Section 5.0). These data can appear anywhere in the input stream after the first title record set. When SEADYN encounters NSFILE>0, these rigid format data are processed into the ship load data file. This is to be done only once in any job deck. It is not possible to stack new problem cases using NEW which intend to define two distinct ship load data files.

SHIP LOAD TITLE RECORD (12A6)*

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|--------------------------|-----------------------|
| 1-72 | SHPCAP | Any descriptive title |

UNIT LABEL RECORD (A6, 4X, A6, 4X, A6, 4X, A6)

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|--------------------------|---|
| 1-6 | WLBL | Wind force label (e.g., "TONS," "POUNDS") |
| 11-16 | CLBL | Current force label |
| 21-26 | LLBL | Length label (e.g., "FEET," "METERS") |
| 1-36 | VLBL | Velocity label (e.g., "KNOT," "FT/SEC") |

NOTE: These labels are output with the ship data as a reminder of the units used. They are used for no other purpose.

SHIP PARAMETERS (8E10.0)

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|--------------------------|------------------------------------|
| 1-10 | TSLT | Total ship length (L) |
| 11-20 | TSAE | End projected wind area (L^2) |
| 21-30 | TSAS | Side projected wind area (L^2) |
| 31-40 | TSWL | Water line length (L) |
| 41-50 | TSB | Beam at midships (L) |
| 51-60 | TSD | Draft at midships (L) |
| 61-70 | TSDSP | Volume displacement (L^3) |
| 71-80 | TSAP | Propeller projected area (L^2) |

*FORTRAN format specification.

WIND RECORD (2I5, 6E10.0)

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|----------------------|---|
| 1-5 | NWIND | Number of wind velocity tables (max is 5) |
| 6-10 | NTHETW | Number of headings in each table (max is 20) |
| 11-20 | SCALE | Test scale factor (A means $1/A^{\text{th}}$ scale) |
| 21-30 | WNDVEL(1) | First wind velocity (smallest) (LT^{-1}) |
| . | . | . |
| 61-70 | WNDVEL(5) | Fifth wind velocity |

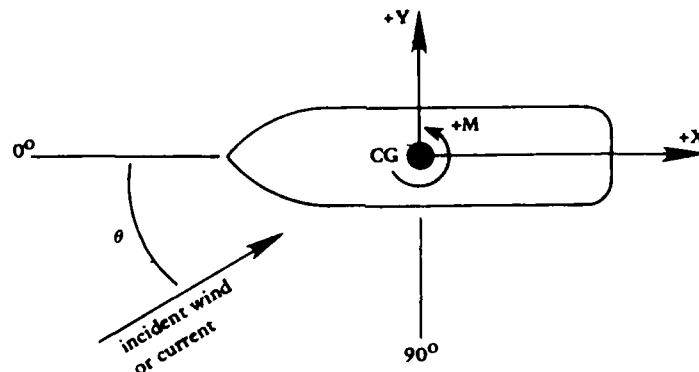
WIND HEADING RECORD(S) (8E10.0)

(Repeat as required to get NTHETW entries)

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|----------------------|------------------------------|
| 1-10 | WNDHED(1) | First wind heading (degrees) |
| 11-20 | etc. | |

NOTES

- Headings should be between 0° and 360° , listed from the smallest to the largest.
- If the largest value is 180° , the loading functions are assumed to be symmetric about 180° for the end forces and skew-symmetric for the side forces and yaw moments.
- The angle is measured relative to the ship's local coordinate system, which is illustrated below:



WIND FORCE RECORD(S) (3E10.0)

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|--------------------------|---|
| 1-10 | WNDCOE(I,1,J) | End force for I th heading and J th velocity (F) |
| 11-20 | WNDCOE(I,2,J) | Side force for I th heading and J th velocity (F) |
| 21-30 | WNDCOE(I,3,J) | Moment for I th heading and J th velocity (FL) |

(I varies before J)

SURFACE CURRENT RECORD(S) (2I5, 6E10.0)

| | | |
|-------|-----------|---|
| 1-5 | NCRNT | Number of current tables |
| 6-10 | NTHETC | Number of headings in each current table |
| 11-20 | TDEPTH | Test water depth (L) |
| 21-30 | CURVEL(1) | First current velocity (smallest) (LT ⁻¹) |
| . | . | |
| . | . | |
| 61-70 | CURVEL(5) | Fifth current velocity |

SURFACE CURRENT HEADING RECORD(S) (8E10.0)

(Repeat as required to get NTHETC entries)

| <u>Columns</u> | <u>Variable Name</u> | <u>Description</u> |
|----------------|--------------------------|---------------------------------|
| 1-10 | CURHED(1) | First Current Heading (degrees) |
| 11-20 | etc. | |

(See notes for WIND HEADING RECORD)

SURFACE CURRENT FORCE RECORD(S) (3E10.0)

(One record for each heading repeated for each velocity)

| | | |
|-------|---------------|---|
| 1-10 | CURCOE(I,1,J) | End force for I th heading and J th velocity (F) |
| 11-20 | CURCOE(I,2,J) | Side force for I th heading and J th velocity (F) |
| 21-30 | CURCOE(I,3,J) | Moment for I th heading and J th velocity (FL) |

(I varies before J)

The ship load file contains one logical record for each ship catalogued on the file, and it is written in a binary form with the following FORTRAN statement:

```
WRITE(10)NWIND, NTHETW, WNDVEL, WNDHED, WNDCOE, SCALE, NCRNT,
NTHETC, CURVEL, CURHED, CURCOE, TDEPTH, TBLOCK, TSLT, TSAE, TSAS,
TSWL, TSB, TSD, TSDSP, TSAP, SHPCAP, WLBL, CLBL, LLBL, VLBL
```

A number of the items in the list are arrays, and they are written in their entirety using the implied DO-LOOP feature of FORTRAN I-O statements. A description of each item, including the dimensions of the arrays, is given below:

| <u>VARIABLE</u> | <u>DESCRIPTION</u> |
|-----------------|--|
| NWIND | Number of wind velocity tables |
| NTHETW | Number of headings in each wind table |
| WNDVEL(5) | Array of wind velocities |
| WNDHED(20) | Array of wind headings |
| WDCOE(20,3,5) | Array of wind load coefficients giving values for up to 20 headings for end force, side force, and yaw moment for up to five wind velocities |
| SCALE | Scale for wind load tests (A means $1/A^{\text{th}}$ scale) |
| NCRNT | Number of current velocity tables |
| NTHETC | Number of headings in each current table |
| CURVEL(5) | Array of current velocities |
| CURHED(20) | Array of current headings |
| CURCOE(20,3,5) | Array of current load coefficients giving values for up to 20 headings for end force, side force, and yaw moment for up to five current velocities |
| TDEPTH | Water depth for test |
| TBLOCK | Ship's block coefficient |

| | |
|------------|---|
| TSLT | Total ship length |
| TSAE | End projected wind area |
| TSAS | Side projected wind area |
| TSWL | Water line length |
| TSB | Beam at midships |
| TSD | Draft at midships |
| TSDSP | Volume displacement |
| TSAP | Propeller projected area |
| SHPCAP(12) | Title of 12 six-character Hollerith words |
| WLBL | Wind force label, six-character Hollerith word |
| CLBL | Current force label, six-character Hollerith word |
| LLBL | Length label, six-character Hollerith word |
| VLBL | Velocity label, six-character Hollerith word |

Appendix B

DESCRIPTION OF EXTERNAL PLOT DATA FILE

A special purpose formatted file can be generated to collect certain of the computed data without saving the entire restart file each time the data is wanted. The main purpose of this file is to provide input for an external (to SEADYN) plotting process. Of course, the data may be extracted from the file for other purposes also. This appendix describes the format of the file and the process used to write it.

The writing of the external plot file is controlled by data fields contained in the OUP data record of an SAO. That record provides pointers to up to three output templates that are defined in the TABL data sets. These templates identify the type of data to be saved and the element and nodes that they are associated with. The frequency of the output to the file is controlled by the OUP data.

The plot file will be generated with the name of 'PLOT.DAT' on file 04. If file 04 has been used for RESTART it will be closed before 'PLOT.DAT' is opened by this statement:

```
1      OPEN(NPLTP,FILE='PLOT.DAT',STATUS='NEW',FORM='UNFORMATTED',  
          ACCESS='SEQUENTIAL')
```

The structure of the plot data file is:

There will be a sequence of flag records and data records. The flag record identifies the form of the succeeding data records. The flag records consist of 3 integer words, a floating point word, and from 1 to 6 character words.

The general form of the flag records is:

```
CHARACTER * 15 AC(6) DOUBLE PRECISION PARM
```

```
READ(NPLTP) ITYPE,INFO,NW,PARM,(AC(1),1-1,NW)
```

There are four types of flag records:

| ITYPE | INFO | NW | PARM | AC | DESCRIPTION |
|--------|--------|----|--------|--------|------------------------------|
| -90000 | 0 | 1 | DTPLOT | dummy | Header record follows |
| -90001 | NNODES | 1 | . | label | NNODES node records follow |
| -90002 | NELTS | NW | . | labels | NELTS element records follow |
| -90003 | NELTS | NW | . | labels | NELTS slope records follow |

In the three data cases the PARM word contains the value of the solution parameter for the records that follow (e.g. time). The labels identify what the data is. These are the same labels that are printed with the optional printout for the selected code. In the element records the value of NW may be from 1 to 6, depending on how many element codes were selected for the template. NW will be 1 (simplex elements) or 2 (catenary elements) for slope records. The catenary elements give the slope at each end.

The following FORTRAN statements show how to read the data for each of the record types.

HEADER RECORD:

```
CHARACTER * 10 HED(8),IDATE,ITIME
CHARACTER * 24 VERSION
READ(NPLTP) HED,IDATE,ITIME,VERSION
```

NODE RECORD:

```
DOUBLE PRECISION X,Y,Z
READ(NPLTP) NODE,ICODE,X,Y,Z
```

ELEMENT RECORD:

```
DIMENSION IELCD(6)
DOUBLE PRECISION ELT(6),IELCD(6)
READ(NPLTP) IELT,(IELCD(I),ELT(I),I=1,NW)
```

SLOPE RECORD:

```
DOUBLE PRECISION ELSLP(3,2)
ICODE=32
DO 100 I=1,NW
100 READ(NPLTP) IELT,(ELSLP(J,I),J=1,3)
```

The following is a short FORTRAN routine that simply reads the plot file and writes what is read. It illustrates how to handle the file.

```
CHARACTER * 15 AC(6)
CHARACTER * 10 HED(8),IDATE,ITIME
CHARACTER * 24 VERSION
DOUBLE PRECISION PARM
DOUBLE PRECISION X,Y,Z
DOUBLE PRECISION ELT(6)
DOUBLE PRECISION ELSLP(3,2)
DIMENSION IELCD(6)
```

*

```

      NPLTP=4
*
      OPEN(NPLTP,FILE='PLOT.DAT',STATUS='OLD',FORM='FORMATTED',
1      ACCESS='SEQUENTIAL')
*
      100 READ(NPLTP,5100,END=300) ITYPE,INFO,NW,PARM,(AC(I),I=1,NW)
      5100 FORMAT(I6,I5,I4,E15.6,6(A15))
           IGO=-ITYPE-90000+1
           WRITE(6,6100) ITYPE,IGO,PARM
      6100 FORMAT(' ITYPE,IGO,PARM',2I7,E15.6)
           GO TO (110,120,130,140),IGO
           STOP
*
      110 READ(NPLTP,5110) HED,IDATE,ITIME,VERSION
      5110 FORMAT(8A10/2A10/A24)
           WRITE(6,6110) HED,IDATE,ITIME,VERSION
      6110 FORMAT(' ',8A10/' ',2I10/' ',A24)
           GO TO 100
*
      120 WRITE(6,6120) AC(1)
      6120 FORMAT(' LABEL',A15)
           DO 121 J=1,INFO
           READ(NPLTP,5121) NODE,ICODE,X,Y,Z
      5121 FORMAT(2I5,3E15.6)
      121 WRITE(6,6121) NODE,ICODE,X,Y,Z
      6121 FORMAT(' NODE,ICODE,X,Y,Z',2I5,3E15.6)
           GO TO 100
*
      130 WRITE(6,6130) (AC(I),I=1,NW)
      6130 FORMAT(' LABELS'/(3X,A15))
           DO 131 J=1,INFO
           READ(NPLTP,5131) IELT,(IELCD(I),ELT(I),I=1,NW)
      5131 FORMAT(I5,6(I3,E12.4))
      131 WRITE(6,6131) IELT,(IELCD(I),ELT(I),I=1,NW)
      6131 FORMAT(' IELT,(CODE,VALUE)',I5,6(I3,E12.4))
           GO TO 100
*
      140 ICODE=32
           DO 220 K=1,INFO
           DO 200 I=1,NW
           WRITE(6,6200) AC(I)
      6200 FORMAT(A15)
      200 READ(NPLTP,5200) IELT,(ELSLP(J,I),J=1,3)
      5200 FORMAT(I5,3E15.6)
           WRITE(6,6220) IELT,(ELSLP(J,I),J=1,3)
      6220 FORMAT(' IELT,SLOPE X,Y,Z',I5,3E15.6)
      220 CONTINUE
           GO TO 100
      300 STOP
      END

```


Appendix C

DEFAULT DRAG COEFFICIENTS

SEADYN provides a set of drag functions that can be used when no specific drag models are available. The functions are described in this appendix. They are based on the work of Choo and Casarella in the early 70's and refined by Choo in 1973. They are essentially regional curve fits of experimental data for smooth cylindrical rods and spheres assuming the independence principle and a simplified form of Morison's equation.

These functions will be used when the user specifies a zero for a drag function code in the drag data set, or specifies nothing for drag function codes on MATE and/or BODY data records. These are called the default functions.

The default coefficients are:

Spherical Bodies

$$\text{Reynolds number, } R_e = \frac{V d}{\nu} = \frac{\text{velocity} \times \text{body diameter}}{\text{kinematic viscosity}} \quad (\text{C-1})$$

$$\begin{aligned} C_D &= 40 \text{ for } R_e \leq 0.1 \\ C_D &= 0.044 + 13.46/(R_e)^{0.5} \text{ for } 0.1 < R_e \leq 1000 \\ C_D &= 0.47 \text{ for } 1000 < R_e \leq 10^5 \\ C_D &= 0.12 \text{ for } R_e > 10^5 \end{aligned} \quad (\text{C-2})$$

Cylindrical Bodies and Cable Elements

$$R_e = \frac{V_N d}{\nu} = \frac{\text{normal velocity} \times \text{body diameter}}{\text{kinematic viscosity}} \quad (\text{C-3})$$

$$R_{eT} = \frac{V_T d}{\nu} = \frac{\text{tangential velocity} \times \text{body diameter}}{\text{kinematic viscosity}} \quad (\text{C-4})$$

$$\begin{aligned}
C_N &= 13 && \text{for } R_e \leq 0.1 \\
C_N &= 0.45 + 5.93/(R_e)^{0.33} && \text{for } 0.1 < R_e \leq 400 \\
C_N &= 1.27 && \text{for } 400 \leq R_e \leq 10^5 \\
C_N &= 0.3 && \text{for } R_e > 10^5
\end{aligned} \tag{C-5}$$

$$\begin{aligned}
C_T &= 10 && \text{for } R_{eT} \leq 0.1 \\
C_T &= 0.598/(R_{eT})^{0.74} && \text{for } 0.1 < R_{eT} \leq 100.55 \\
C_T &= 0.017 && \text{for } R_{eT} > 100.55
\end{aligned} \tag{C-6}$$

$$\begin{aligned}
D_N &= \frac{1}{2} \rho C_N V_N^2 \\
D_T &= \frac{1}{2} \rho C_T \pi d V_T^2
\end{aligned} \tag{C-7}$$

where: D_N = normal drag per unit length
 D_T = tangential drag per unit length
 ρ = fluid mass density
 d = cable diameter
 V_N, V_T = normal and tangential components of velocity, respectively

C_N, C_T = normal and tangential coefficients, respectively

Note that the default coefficients assume smooth cylindrical cable. Other cable constructions (chain, faired cable, etc.) will require alternate coefficients and possibly alternate loading functions (see for example, References 1 and 21). These can be incorporated into SEADYN via the user-defined subroutine USRDRG (consult Appendix D) or the DRAG data set (consult Section 6.2.3).

Appendix D

USER-SUPPLIED SUBROUTINES

The SEADYN program allows additional modeling flexibility by providing for three user-defined subroutines. They are: USRDRG, USRTFN, USRCUR, and USRLIN. Be sure to assign all floating point values to double precision when a double precision version of SEADYN is being used.

The USRDRG subroutine provides for definition of line and lumped body drag coefficients. The routine is called each time the coefficient is required. The necessary subroutine parameter definitions are:

```
SUBROUTINE USRDRG (IBOD, IDR, RE, RET, CN, CT, DRGDAT)
DIMENSION DRGDAT (20)
```

where: IBOD = Body type code set by calling routine

- 1 - spherical buoy
- 2 - cylindrical buoy
- 3 - cable

IDR = drag function code
passed from the DRAG data set with the negative sign removed (i.e., $IDR > 0$)

RE = Reynolds number based on the present normal component of the computed relative fluid velocity

RET = Reynolds number based on the present tangential component of the computed relative fluid velocity (not given for spheres)

CN = return variable for the calculated normal drag coefficient

CT = return variable for the calculated tangential drag coefficient (not used for spheres)

DRGDAT = the optional drag function parameters input with the DRAG data record for this drag function

The data parameters, IBOD and IDR, can be used to select the particular functions from user-defined catalogs of functions. The DRGDAT parameters allow the user to pass specific data to the coefficient computation from the input data.

The USRTFN provides a single-valued function in time that defines the time variation of loads, currents, motions, payout, etc. The necessary subroutine parameter definitions are:

```
SUBROUTINE USRTFN (T, F, N, TPARM)
  DIMENSION TPARM(20)
```

where: T = current time, $T \geq 0$

F = the returned value of the time function

N = the time function code, the absolute value of the numbers provided in the TFUN data set.

TPARM = a single-dimensioned array of user-defined input parameters provided for this function in the TFUN data set. The maximum number of parameters is 20.

The USRCUR subroutine is called to define the fluid velocity at all node points in the structure when the FLOW library indicates a user-defined routine. Unless signaled otherwise, the routine is called at every iteration or step of the subanalysis. DYN and TSSS subanalyses can call USRCUR to get the space-dependent flow components and use the TFUN library to define time variation or can require USRCUR to give both time and space variation. (See the FLOW library data.) The necessary subroutine parameter definitions are:

```
SUBROUTINE USRCUR (T, N, NN, X, V, FLPAR, NODKEY)
  DIMENSION X(*), Y(*), NODKEY (8,*) FLPAR (40)
```

where: T = current time, $T \geq 0.0$

N = the flow field code, the absolute value of the number provided in the FLOW data set

NN = number of nodes

X = nodal positions (X, Y, Z position for each node)

V = nodal flow vector (X, Y, Z velocity for each node). These values for all NN nodes are to be returned at each call.

FLPAR = a single-dimensioned array of user-defined input parameters provided for this function in the FLOW data set. The maximum number of parameters is 40.

NODKEY = is an array of nodal feature information

Since the SEADYN internal data structure allows for a variable number of components per node, it is not possible to address nodal data as a two dimensional array. The NODKEY gives the following data about each node:

NODKE (I,J) J represents the node number

I Description:

- 1 Internal equation number reference. This is the position in the nodal data vector just prior to the position where the data for the Jth node begins. For example: NODKEY(1,8) will contain the number NREF such that X(NREF+1) gives the first component of X data for node 8.
- 2 The number of degrees of freedom assigned to this node.
- 3 Local coordinate system code for displacements at this node.
- 4 Master node number for this node.
- 5 Flag signalling there is a vertical tangent at this node.
- 6 Local coordinate system flag for forces at this node.
- 7 Limit set number this node has been assigned.
- 8 Payout set code, the payout data set number this node is assigned to.

Normally only the I=1 values will be used. The values in the X(*) array may be used to evaluate spatially varying flow fields. Be careful not to alter the values in any of the parameters except the V(*) vector. Define NREF = NODKEY(1,J) for J=1,NN. Assign the x, y, z velocity components to the V(NREF+1), V(NREF+2), and V(NREF+3) positions, respectively.

The USRLIN subroutine provides element drag loads based on a general drag model that uses the free-stream velocity and angle of attack, rather than the independence principle. It will be called at each solution iteration for each element using material data that identifies the drag function with a negative number having a magnitude greater than 99. The subroutine must be written to return the three components of drag load per unit length for the velocity and angle given and in the coordinate system indicated by DRGPAR(2). The necessary subroutine parameter definitions are:

```
SUBROUTINE USRLIN (IGO,ILT,KEND,VEL,VIS,D,HD,PHID,DS,DRGPAR,  
                  CN,CS,MOROUT,FNODL)
```

```
DIMENSIONS DS(*), DRGPAR(20), FNODL(3)
```

```
IGO    = drag function type code (100 or greater)  
ILT    = element number  
KEND   = element end (1 or 2)  
VEL    = magnitude of free-stream velocity  
VIS    = fluid kinematic viscosity  
D      = element drag diameter  
HD     = 1/2 rho (half density for fluid)
```

PHID = line inclination angle (degrees) relative to flow
 DS = element lengths
 DRGPAR = drag function parameters
 (1) alternate characteristic length
 (2) local coordinate system code
 0 for TANGENTIAL, NORMAL, SIDE
 1 for DRAG, LIFT, SIDE
 (3) drag function number for CN
 (4) drag function number for CS
 remainder at disposal of user. (5 to 20)
 CN = precomputed normal drag coef (PHID=90)
 CS = precomputed side force coef
 MOROUT = element connection regularity flag
 0 means regular sequential connectivity
 -1 means irregularity exists, must treat elements
 individually, not as a regular sequence from one end to
 the other.
 FNODL = local components of node load function (to be computed
 by this routine)

Appendix E

RESTART FILE STRUCTURE

The SEADYN program creates up to three restart files (one each for the DEAD, LIVE, and DYN subanalyses). Multiple selections of a sub-analysis type simply extends the file unless a rewind is signaled on the SAVE data record. A counter is provided for each of the files to keep track of how many restart records have been written. The FORTRAN file codes used are:

- 01 - DEAD
- 02 - LIVE (and TSSS)
- 03 - DYN

Each time the file is rewound, the counter for that file is set to zero, and a label record is written. The write statement is:

```
WRITE (NFILE) (TITLE (I), I = 1, NHED), NINA, NIBASE, NPRECZ,VERSION
```

where: TITLE = page heading title card for the run (Each word is assumed to have 10 characters.)

NINA = size of unlabeled common when the file is saved

NPRECZ = precision number for floating point numbers

1 = single precision

2 = double precision

NHED = the number of words in the title = 8

Each restart save operation uses the following write statement:

```
WRITE(NTAPE)NFILE,(IA(I),I=1,NIBASE),(ACOM(I),I=1,NINA)
WRITE(NTAPE)(B(I),I=1,NINB),(C(I),I=1,NINC),
+(RL(I),I=1,NINRL),(P(I),I=1,NINPO),(PL(I),I=1,NINPLT),
+(T(I),I=1,NINT),(SH(I),I=1,NINSHP),(STM(I),I=1,NINSTM)
WRITE(NTAPE)
. (IAABU(I),I=1,NINBI),(IAACA(I),I=1,NINCI)
. ,(IAACL(I),I=1,NINCLI),(IPL(I),I=1,NINIPL)
. ,(IAAPO(I),I=1,NINPOI),(IAASHP(I),I=1,NINSHI)
. ,(IAASTM(I),I=1,NINSTI),(IAATIM(I),I=1,NINTMI)
+,DLD,WLD,DYN,CHECKR,NOVEL,NOITER,NOFIUD,NOLOAD,FEEDBK,POUT,
+ REFUP,STEPUP,RBDYFL,BTMFOP
IF(RBDYFL) WRITE(NTAPE) (RB(I),I=1,NINRBY),(IRR(I),I=1,NINRBI)
. ,(CD(I),I=1,NINCDY),(ICDY(I),I=1,NINIDY)
```

where the arrays are defined by:

```
COMMON/ACOM/ A(1)
COMMON/IACOM/ IA(1)
COMMON/BUOYS/ B(1)
COMMON/CABLE/ C(1)
COMMON/CDYNTD/ CD(1)
COMMON/CONTRL/ RL(1)
COMMON/PAYOUT/ P(1)
COMMON/RGDBDY/ RB(1)
COMMON/TIMED/ T(1)
COMMON/SHIPS/ SH(1)
COMMON/STRUM/ STM(1)
COMMON/IBUOYS/ IAABU(1)
COMMON/ICABLE/ IAACA(1)
COMMON/IDYNTD/ ICDY(1)
COMMON/ICNTRL/ IAACL(1)
COMMON/IPAYOT/ IAAPO(1)
COMMON/IRGBDY/ IRB(1)
COMMON/ISHIPS/ IAASHP(1)
COMMON/ISTRUM/ IAASTM(1)
COMMON/ITIMED/ IAATIM(1)
COMMON/PLTDAT/ PL(1)
COMMON/IPLTDT/ IPL(1)
```

```
COMMON/LOGIC/DLD,WLD,DYN,CHECKR,NOVEL,NOITER,NOFLUD,NOLOAD,
1 FEEDBK,POUT,REFUP,STEPUP,RBDYFL
```

The sizes of the arrays are given by:

```
COMMON/SIZE/ NINA,NINB,NINC,NINPLT,NINPO,NINRL,NINSHP,NINSTM,NINT,SIZE 3
. NCOM,NICOM,IFILE(4),NPRECZ,NINBI,NINCI,NINCLI,NINIPL,          SIZE 4
. NINPOI,NINSHI,NINTMI,NINRBY,NINRBI,NINCDY,NINIDY,NIBASE,NINSTI SIZE 5
```

These sizes are identified in Reference 2, and count the number of words to be read/written. The actual contents of the labeled common blocks are defined in the calling program.

Appendix F

SHIP'S WIND AND CURRENT LOADING FUNCTIONS

This Appendix describes the tabular approach used in SEADYN to obtain static loads on a ship subjected to winds and surface currents.

The ship loads are assumed to be applied through the center of gravity of the ship. Three sources of loading are considered: wind loads, surface current loads, and point loads representative of working loads. The point loads are specified through the normal loading options (see LOAD record). It is assumed that the point loads do not change their magnitude or global directions as the ship moves to a new position. Specification of the wind and current loadings is somewhat more complicated. The SEADYN program provides two approaches to defining these loads. The first approach is in the form of loading tables that give loads versus ship's heading relative to the flow. The second approach uses approximate analytical expressions, which are described in Appendix G.

The tabular approach is based on the procedures given in NAVFAC's Design Manual 26 (Ref 17). The DM-26 approach utilizes experimental measurements for the forces and moments for various headings of wind and current for a set of "representative" vessels. Similarity scaling is then applied to get loading values for ships other than the test models.

The DM-26 procedure begins with a set of load measurements obtained from subscale tests on a representative ship's model or any other available source. The measurements give values for the lateral and longitudinal forces and yaw moment versus flow heading and flow velocity. These measurements represent the combined effects of such phenomena as profile and friction drag, lift-induced side forces, and shifts in the center of pressure. Tables of these measurements can be specified as either input to the program or as a special ship loading file previously generated and saved for subsequent referencing by SEADYN. This saved file is assumed to be on logical unit 08 (see Appendix I).

Given the headings of wind and surface current relative to the ship, the loads are obtained by linear interpolation in the tables. In the event that there are tables provided for more than one velocity, the table for the velocity nearest the one specified in the analysis will be used. This is determined by comparing the squares of the velocity ratios.

After the load coefficients are obtained from the tables for the given heading, they must be scaled to account for differences in the conditions modeled in the test and for those being analyzed. The scaling accounts for differences in flow velocity, water depth, and ship geometry. The formulas for adjusting for these effects are given in DM-26 and are restated here for completeness.

WIND

$$F_s = C_f V^2 F_{ms} (A_s/A_{ts}) \quad (F-1)$$

$$F_e = C_f V^2 F_{me} (A_e/A_{te}) \quad (F-2)$$

$$M_w = C_m V^2 M_m (A_s/A_{ts}) (L/L_t) \quad (F-3)$$

where: F_s = lateral force on ship

F_e = longitudinal force on ship

M_w = yawing moment on ship

F_{ms} = lateral force on model

F_{me} = longitudinal force on model

M_m = yawing moment on model

V = wind velocity

A_s = side-projected area above the water line of ship being analyzed

A_{ts} = side-projected area above the water line of modeled ship

A_e = end-projected area above the water line of ship being analyzed

A_{te} = end-projected area above the water line of modeled ship

L = length of ship being analyzed

L_t = length of modeled ship

$$C_f = \frac{S^2}{V_T^2} \quad (F-4)$$

$$C_m = \frac{S^3}{V_T^2} \quad (F-5)$$

S = linear scale of the model (e.g., 50 to 1; $S = 50$)

V_T = wind velocity used in model test

CURRENT

$$h_2 = h_1 I_{W2}/I_{W1} \quad (F-6)$$

$$V_1 = V_2 (I_{W1}/I_{W2})^{1/2} \quad (F-7)$$

$$F_2 = F_1 \Delta_2/\Delta_1 \quad (F-8)$$

$$M_2 = M_1 (\Delta_2/\Delta_1)(I_{W2}/I_{W1}) \quad (F-9)$$

where: h = depth of water

V = velocity of current

I_W = water line length of vessel

F = lateral or longitudinal resisting force

Δ = displacement

M = yaw resisting moment

Subscript 1 denotes the full-scale vessel for which the model test was made, and subscript 2 denotes the vessel being analyzed.

When the velocity from Equation F-7 does not correspond to one of the tables given for the model test, then the forces and moments must be selected from the tables corresponding to the velocity nearest the value of V_1 in Equation F-7. It will then be necessary to adjust the values by the square of the ratio of the V_1 velocity and the velocity represented in the tables, V_{t1} .

It is quite likely that the depth at the proposed mooring site will not be the same as that obtained for h_2 in Equation F-6. In that event, a correction for depth is required. DM-26 suggests that the correction be made assuming an inverse relationship with the side resistances at the two depths in question. The curves given in Graph 124 (EC-2) of DM-26 are used along with Equation F-6 for this purpose. The data are given in tabular form and the side resistances are obtained by logarithmic interpolation. The resistance for a depth greater than that in the table will be the last value in the table.

The adjustments for current velocity and depth are summarized by the following equations:

$$F'_{s2} = \left[r_h \frac{V_1^2}{V_{t1}^2} F_{s2} \right] \quad (F-10)$$

$$F'_{e2} = \left[f_h \frac{\Delta_2}{\Delta_1} \frac{V_1^2}{V_{t1}^2} \right] \left[F_{e1} - \frac{1}{2} \rho C_p A V_{t1}^2 \right] \quad (F-11)$$

$$+ \frac{1}{2} \rho C_p A V_2^2$$

$$M'_2 = \left[f_h \frac{V_1^2}{V_{t1}^2} M_2 \right] \quad (F-12)$$

where: f_h = the depth scaling factor

V_{t1} = the velocity at which the test data was obtained

A = the propeller projected area

C_p = the propeller drag coefficient

ρ = fluid density

The primes indicate the value has been adjusted to the desired conditions for the mooring site. Equation F-11 reflects the adjustment in the longitudinal force recommended by DM-26 with the assumption that $(1/2) \rho C_p = 2.88$ (with V in knots). Assuming the specific weight of seawater is 64 lb/ft^3 and the acceleration due to gravity is 32.2 ft/sec^2 , then $C_p = 1.00$. The form using $(1/2) \rho C_p$ rather than 2.88 is required to make the procedure dimensionally independent.

Appendix G

BUILT-IN LOAD FUNCTIONS FOR SHIPS

This Appendix describes the analytical approach for obtaining static loads on a ship subjected to winds and surface currents. The approximate analytical expressions for ship's loading are based primarily on the work of Hughes (Ref 18), standard Naval architectural formulas (Ref 19), and Altman (Ref 20).

It should be emphasized that these analytical expressions are to be viewed as a convenient alternative to the DM-26 experimental curve procedure. It remains to be demonstrated that they are capable of giving reliable approximations of the ship's loading.

The wind loading is given by:

$$F = K \rho_a V^2 (A_s \sin^2 \theta + A_e \cos^2 \theta) \cos (\alpha - \theta) \quad (G-1)$$

where: K = constant, 0.6

F = resultant wind force

ρ_a = mass density of air

V = wind velocity

θ = wind heading relative to the bow

α = heading of the resultant wind force relative to the bow

A_s = side projected area of ship above water line

A_e = end projected area of ship above water line

The heading of the resultant wind force, α , is approximated as a function of θ in a 7th order polynomial as follows:

$$\begin{aligned} \alpha = & 0.0715608 + 7.954381 \theta - 0.3254561 \theta^2 \\ & + 0.0073131 \theta^3 - 9.3966 \times 10^{-5} \theta^4 \\ & + 6.85008 \times 10^{-7} \theta^5 - 2.6323 \times 10^{-9} \theta^6 \\ & + 4.1453 \times 10^{-12} \theta^7 \end{aligned} \quad (G-2)$$

In Equation G-2 both θ and α are measured in degrees.

The distance between the ship forward perpendicular and the center of wind pressure, X_{cp} , can be approximated as a polynomial function of the wind direction, θ . This relationship is:

$$\begin{aligned} \frac{X_{cp}}{L} = & 0.2004112 + 0.0048641 \theta - 4.52442 \times 10^{-5} \theta^2 \\ & + 5.45736 \times 10^{-7} \theta^3 - 3.78789 \times 10^{-9} \theta^4 \\ & + 1.02881 \times 10^{-11} \theta^5 \end{aligned} \quad (G-3)$$

Here, as above, θ is measured in degrees. Also, L is the overall ship length. The yawing moment due to wind is then approximated by:

$$M_W = F L \sin \alpha \left(\frac{1}{2} - \frac{X_{cp}}{L} \right) \quad (G-4)$$

Analytical expressions for the resistances from current effects utilize the approach presented by Altman in Reference 20. These expressions are summarized below:

$$F_s = F_{s\infty} \left(1 + \frac{10}{(h/H)^2 - 1} \right) \quad (G-5)$$

$$F_{s\infty} = 0.215 \rho_w V^2 L_w H \sin \theta \quad (G-6)$$

$$F_e = \frac{1}{2} \rho_w V^2 (S_w C_R + A_p C_p) \cos \theta \quad (G-7)$$

$$M = F_s L_{CP} \quad (G-8)$$

where: F_s = lateral current force at the specified water depth

$F_{s\infty}$ = lateral current force in deep water

F_e = longitudinal current force

m = yaw current moment

∇ = displaced volume

C_s = wetted surface coefficient, input on SHIP record

C_R = hull resistance coefficient, input on SHIP record
or calculated as $C_r + C_f + 0.0005$ (G-9)

C_r = residuary resistance coefficient (see following discussion)

$$C_f = \text{frictional resistance coefficient,}$$

$$= \frac{0.456}{(\log_{10} R_e)^{2.58}} - \frac{1,700}{R_e} \quad R_e \geq 5 \times 10^5 \quad (G-10)$$

$$= 0.002 \quad R_e < 5 \times 10^5$$

$$S_w = C_s \nabla L_w \quad (G-11)$$

R_e = Reynolds number for the hull (based on longitudinal component of flow and ship length)

$$L_{CP} = \text{distance from midships to hull center of pressure}$$

$$= L[\bar{L}_{90} + 0.00226 (\theta - 90^\circ)] \quad \text{for } 0^\circ \leq \theta \leq 180^\circ$$

$$= L[\bar{L}_{90} + 0.00226 (\theta - 270^\circ)] \quad \text{for } 180^\circ \leq \theta \leq 360^\circ \quad (G-12)$$

\bar{L}_{90} = ratio of distance to center of pressure at $\theta = 90^\circ$ to the distance to the center of hull side area

h = water depth

H = ship draft

ρ_w = mass density of water

L_w = waterline length of ship

A_p = propeller projected area

C_p = propeller drag coefficient

Several of these terms require further discussion. The hull resistance coefficient, C_R , represents the sum of various coefficients for different sources of hull resistance. This coefficient can be input or calculated in the computer program. When no input is given for C_R , it will be calculated as the sum of a residuary resistance coefficient, a frictional resistance coefficient, and a fouling/surface effect coefficient. The fouling/surface effect coefficient is given an arbitrary value of 0.0005. The frictional coefficient is calculated from Equation G-10 and the residuary coefficient is obtained from linear interpolation of a digitized form of Figure 38 of Reference 20. This method of obtaining C_R is limited to low flow velocities since wave-making resistances are ignored.

The longitudinal location of the center of pressure for a hull skewed with respect to the flow is estimated by Equation G-12. This requires an estimate of the ratio of the distance to the center of pressure and the center of area for beam flow, \bar{L}_{90} . This factor is estimated by linear interpolation between the values for the ship's DD-692 and EC-2 using the block coefficient as a reference. Reference 20 gives the values for \bar{L}_{90} or the DD-692 and EC-2 as 0.056 and -0.138, respectively. (Negative means aft of midships.)

Appendix H

MOORING COMPONENT INVENTORY

The mooring component inventory contains data tables for the following:

Anchors:

Navy standard stockless
NAVSHIPS lightweight
NAVFAC STATO

Buoys:

Bar riser chain type

Chain:

Steel stud-link

Hawsers:

Samson braids -- 2-in-1[®] Nylon
 2-in-1[®] Power Braid
 2-in-1[®] Stable Braid[®]
 12 Strand Blue Streak[®]

The inventory lists weights, buoyancies, and strengths in pounds. Lengths and buoy dimensions are in feet. Hawser and chain sizes are in inches. These units may be converted to those needed in the analysis by providing the appropriate conversion factors on the INVE record. The contents of the inventory can be obtained by setting the print flag in Word (4) of the INVE record. The listing of the present inventory is presented below.

C O M P O N E N T I N V E N T O R Y

ANCHOR TYPE = NAVY STD STOCKLESS

| WEIGHT | FED. STOCK NO. | HOLD. POWER (FIRM SAND) |
|------------|----------------|----------------------------|
| .30000E+04 | C2040-516-7758 | .21000E+05 |
| .50000E+04 | C2040-516-7757 | .35000E+05 |
| .60000E+04 | C2040-516-7756 | .42000E+05 |
| .70000E+04 | C2040-516-7755 | .49000E+05 |
| .90000E+04 | C2040-516-7754 | .63000E+05 |
| .10000E+05 | C2040-272-2244 | .70000E+05 |
| .13000E+05 | C2040-272-2245 | .91000E+05 |
| .14500E+05 | C2040-272-2246 | .10150E+06 |
| .18000E+05 | C2040-516-7753 | .12600E+06 |
| .20000E+05 | C2040-272-2247 | .14000E+06 |
| .25000E+05 | C2040-272-2242 | .17500E+06 |
| .30000E+05 | C2040-272-2243 | .21000E+06 |
| .40000E+05 | C2040-277-2423 | .28000E+06 |

ANCHOR TYPE = NAVSHIP(LWT)

| WEIGHT | FED. STOCK NO. | HOLD. POWER (FIRM SAND) |
|------------|----------------|----------------------------|
| .10000E+03 | H2040-377-8600 | .28374E+04 |
| .15000E+03 | H2040-377-8601 | .39565E+04 |
| .20000E+03 | H2040-377-8602 | .50091E+04 |
| .30000E+03 | H2040-377-8603 | .69848E+04 |
| .50000E+03 | H2040-377-8604 | .10619E+05 |
| .75000E+03 | H2040-377-8605 | .14807E+05 |
| .10000E+04 | H2040-377-8606 | .18746E+05 |
| .15000E+04 | H2040-377-8607 | .26140E+05 |
| .20000E+04 | H2040-377-8608 | .33095E+05 |
| .25000E+04 | H2040-377-8609 | .39740E+05 |
| .30000E+04 | H2040-377-8610 | .46148E+05 |
| .40000E+04 | H2040-377-8611 | .58426E+05 |
| .50000E+04 | H2040-378-5633 | .70157E+05 |
| .60000E+04 | H2040-378-5634 | .81470E+05 |
| .10000E+05 | H2040-377-8612 | .12385E+06 |
| .13000E+05 | H2040-377-8613 | .15358E+06 |

ANCHOR TYPE = NAVFAC STATO

| WEIGHT | FED. STOCK NO. | HOLD. POWER (FIRM SAND) |
|------------|------------------|----------------------------|
| .20000E+03 | 2CF2040-800-9659 | .40000E+04 |
| .30000E+04 | 2CF2040-702-7864 | .60000E+05 |
| .60000E+04 | 2CF2040-702-6785 | .12000E+06 |
| .90000E+04 | 2CF2040-702-6786 | .18000E+06 |
| .12000E+05 | 2CF2040-702-6787 | .24000E+06 |
| .15000E+05 | 2CF2040-801-7938 | .30000E+06 |

STEEL STUD-LINK CHAIN

| SIZE | STRENGTH | WEIGHT/LENGTH |
|--------|----------|---------------|
| .7500 | 48550. | 5.5556 |
| .8750 | 65280. | 7.7778 |
| 1.0000 | 84500. | 9.4444 |
| 1.1250 | 106080. | 12.2222 |
| 1.2500 | 130070. | 15.0000 |
| 1.3750 | 156330. | 17.7778 |
| 1.5000 | 185060. | 21.1111 |
| 1.6250 | 216030. | 24.4444 |
| 1.7500 | 249210. | 28.3333 |
| 1.8750 | 284540. | 32.7778 |
| 2.0000 | 322000. | 36.6667 |
| 2.1250 | 361530. | 41.1111 |
| 2.2500 | 403100. | 46.6667 |
| 2.3750 | 446660. | 51.6667 |
| 2.5000 | 492190. | 57.7778 |
| 2.6250 | 539620. | 63.3333 |
| 2.7500 | 588930. | 70.0000 |
| 2.8750 | 640070. | 76.6667 |
| 3.0000 | 693000. | 83.3333 |
| 3.1250 | 747680. | 91.1111 |
| 3.2500 | 804070. | 98.3333 |
| 3.3750 | 862130. | 106.1111 |
| 3.5000 | 921810. | 114.4444 |
| 3.6250 | 983080. | 122.7778 |
| 3.7500 | 1045900. | 131.1111 |
| 3.8750 | 1110210. | 140.0000 |
| 4.0000 | 1176000. | 148.8889 |
| 4.1250 | 1234200. | 158.8889 |
| 4.2500 | 1311790. | 170.0000 |
| 4.3750 | 1381330. | 183.8889 |
| 4.5000 | 1452930. | 198.3333 |

B U O Y D A T A

BUOY TYPE = BAR RISER CHAIN

| O. D. | HEIGHT | WEIGHT | NOM. | BUOYANCY | MAX. | BUOYANCY | FED. STOCK NO. |
|----------|---------|--------|------|----------|------|----------|-----------------|
| 6.53125 | 4.03125 | 2200. | | 3562. | | 4300. | C2050-223-3657 |
| 7.03125 | 5.03125 | 2500. | | 5835. | | 7518. | C2050-264-4497 |
| 9.50000 | 5.00000 | 7700. | | 7420. | | 10445. | C2050-223-3665 |
| 10.50000 | 6.50000 | 9600. | | 14414. | | 20879. | C2050-223-3662 |
| 10.50000 | 7.50000 | 10100. | | 17608. | | 25921. | C2050-264-44981 |

H A W S E R D A T A

| SIZE | 2-IN-1 NYLON | | 2-IN-1 POWER BRAID | | 2-IN-1 STABL BRAID | | 12 ST. BLUESTREAK | |
|--------|--------------|-------------|--------------------|-------------|--------------------|-------------|-------------------|-------------|
| | STRENGTH | WEIGHT/100L | STRENGTH | WEIGHT/100L | STRENGTH | WEIGHT/100L | STRENGTH | WEIGHT/100L |
| .5000 | 8300. | 6.60 | 7500. | 5.90 | 7500. | 7.90 | 6300. | 6.40 |
| .7500 | 18000. | 15.00 | 16000. | 13.00 | 16000. | 18.00 | 13600. | 14.00 |
| 1.0000 | 31300. | 26.00 | 28400. | 24.00 | 28400. | 32.00 | 23500. | 25.00 |
| 1.0625 | 36500. | 31.00 | 33200. | 28.00 | 33200. | 37.00 | 0. | .10 |
| 1.1250 | 42000. | 36.00 | 38000. | 32.00 | 38000. | 43.00 | 31500. | 33.00 |
| 1.2500 | 47700. | 41.00 | 43600. | 37.00 | 43600. | 49.00 | 36000. | 38.00 |
| 1.3125 | 54000. | 47.00 | 49000. | 42.00 | 49000. | 56.00 | 40600. | 44.00 |
| 1.5000 | 67500. | 60.00 | 61400. | 53.00 | 61400. | 71.00 | 50800. | 55.00 |
| 1.6250 | 82600. | 74.00 | 75000. | 66.00 | 75000. | 88.00 | 62200. | 68.00 |
| 1.7500 | 99000. | 89.00 | 90000. | 80.00 | 90000. | 106.00 | 74400. | 82.00 |
| 2.0000 | 117000. | 106.00 | 106000. | 95.00 | 106000. | 126.00 | 88000. | 99.00 |
| 2.1250 | 136000. | 124.00 | 124000. | 111.00 | 124000. | 148.00 | 102000. | 115.00 |
| 2.2500 | 156000. | 144.00 | 142000. | 129.00 | 142000. | 172.00 | 117000. | 134.00 |
| 2.5000 | 178000. | 165.00 | 162000. | 148.00 | 162000. | 197.00 | 134000. | 153.00 |
| 2.6250 | 202000. | 188.00 | 183000. | 168.00 | 183000. | 224.00 | 151000. | 175.00 |
| 2.7500 | 226000. | 212.00 | 205000. | 190.00 | 205000. | 253.00 | 170000. | 197.00 |
| 3.0000 | 252000. | 238.00 | 229000. | 213.00 | 229000. | 284.00 | 190000. | 222.00 |
| 3.2500 | 308000. | 294.00 | 280000. | 263.00 | 280000. | 350.00 | 232000. | 274.00 |
| 3.6250 | 369000. | 356.00 | 336000. | 318.00 | 336000. | 424.00 | 278000. | 332.00 |
| 4.0000 | 436000. | 423.00 | 396000. | 379.00 | 396000. | 504.00 | 327000. | 395.00 |
| 4.2500 | 504000. | 497.00 | 461000. | 444.00 | 461000. | 592.00 | 381000. | 463.00 |
| 4.6250 | 586000. | 576.00 | 531000. | 515.00 | 531000. | 686.00 | 439000. | 537.00 |
| 5.0000 | 666000. | 662.00 | 606000. | 592.00 | 606000. | 788.00 | 500000. | 617.00 |

Appendix I

DESCRIPTION OF THE EXTERNAL MOTION DATA FILE

The boundary movement option, MOVE, allows the definition of node point movement through an externally generated data file. The contents of that file is a table giving the value of nodal position (or displacement) versus time. That table may be assigned to a TFUN time sequence data set and then referenced by the MOVE record. The file is read each time the option IFCOD=6 is selected in the TFUN data definition.

Only one external motion data file may be selected on a run, but it may be accessed repeatedly (limited by the maximum number of time sequence functions allowed).

The file name to be assigned to this data set is "NODMOV.TAB" (see Appendix J).

The first record is an alphanumeric title (80A1). Succeeding records have no limit, but only 500 may be selected in any TFUN table. The format is

TIME, X,Y,Z (4E15.6)

The point positions can be defined in a convenient local right-handed cartesian system. The TFUN option, ITFCOD=6, provides for the specification of Euler angles to relate this local system to the problem global system. That option also provides for time shifting and units conversion.

Appendix J

PROGRAM SIZE LIMITATIONS AND STORAGE REQUIREMENTS

An attempt has been made to make the SEADYN program flexible in the size of the problems it can handle. There are, however, some specific array size limits coded into the program. These have been chosen large enough to accommodate most models, and can be increased by modifying the specific dimension statements and size variables. This requires program recompilation and assembly, the description of which is beyond the scope of this manual.

The specific size limitations are:

| | | |
|--------------------|--|-----|
| Maximum number of: | bodies in BODY record | 50 |
| | body locations given by BLOC record | 50 |
| | limit conditions in LIMJ record | 50 |
| | limit locations given by BLOC record, etc. | 250 |
| | elements connecting to a limited node | 10 |
| | cable materials in MATE record | 10 |
| | entries in any tension/strain table | 20 |
| | entries in FLUI record | 2* |
| | catenary lines of nodes generated by NGEN record | 200 |
| | ship/platform rigid bodies defined by SHIP record | 5 |
| | payout/reel-in points in DYN or TSSS SAO | 5 |
| | master nodes | 100 |
| | moved components in DEAD/LIVE/MODE SAO | 30 |
| | moved nodes defined in DYN or TSSS SAO | 5 |
| | lines connecting to a node where an anchor holding power CHEK is made | 20 |
| | strum strings defined by STRUM record | 30 |
| | elements in any strum string | 20 |
| | flow fields defined by FLOW record | 10 |
| | parameters associated with any flow field | 10 |
| | drag functions defined by DRAG record | 20 |
| | time functions defined by TFUN record | 20 |
| | parameters associated with any time or drag function | 20 |
| | load variation sets defined by LOAD/LVAR records | 3 |
| | PROB + REST data sets in any run | 250 |
| | rigid format data sets in any run | 1* |
| | externally defined nodal movement tables | 5 |

*Program logic limitation

| | |
|--|-----|
| time points in each nodal movement table | 500 |
| wind velocities on ship load file | 5 |
| wind headings on ship load file | 20 |
| current velocities on ship load file | 5 |
| current headings on ship load file | 20 |

The size restrictions related to the number of nodes and line elements that can be included in any model are not rigidly defined by dimension statements. A form of variable dimensions is used, which takes a given block of storage and partitions it according to the problem size and specific needs of each analysis option. The main program for SEADYN is simply a routine that defines the size of common and calls the controlling routine. The minimum storage required for a problem is controlled by the number of nodes and number of elements defined. Two common blocks are used to store the variable data. These are /ACOM/ and /IACOM/. All of the node and element dependent data that has a floating point form is managed in /ACOM/. The related fixed point data is managed in /IACOM/. The number of data words needed as a base for problem execution is given by:

$$\begin{aligned} \text{NN3} &= 3*\text{NN} + 3*\text{MXSHIP} + 3*\text{MXBLOC} \\ \text{NBASE} &= 38*\text{NE} + 21*\text{NN3} + 9*\text{NN} && \text{for /ACOM/} && (\text{I-1}) \\ \text{NIASE} &= 8*\text{NE} + \text{NN3} + 8*\text{NN} && \text{for /IACOM/} \end{aligned}$$

Where: NN = number of nodes
 NE = number of elements
 MXSHIP = maximum number of ships allowed (5)
 MXBLOC = maximum number of body locations allowed (50)

The values for these base storage requirements are printed with PROBLEM summary output on each run.

Additional storage is required by each of the analysis options. The formulas used to calculate storage needed for each SAO type are:

DEAD, LIVE, TSSS, DYN:

$$\text{NEED} = \text{NBASE} + \text{NF3}*\text{IB} \quad (\text{I-2})$$

MODE:

$$\text{NEED} = 0.5*(3*\text{NF3}*\text{NF3} + 7*\text{NF3}) \quad (\text{I-3})$$

CHEK:

$$\text{NEED} = \text{NBASE} \quad (\text{I-4})$$

where: IB = equation half bandwidth (0 for DYN - DIM solutions)

$$\text{NF3} = 3*\text{NN} + 3*\text{NSHIPS} + 3*\text{NMBUO}$$

NSHIPS = number of ships rigid bodies in model
 NMBUO = number of mooring buoy type rigid bodies in model

Storage size checks are made at the beginning of each SA0 to determine if enough space is available. If not, a message is printed to indicate the space needed, and the run is aborted.

The major users of storage space are simultaneous equation solvers and the Jacobi eigenvalue solver. These analyses are done entirely in main memory assuming sparse and banded matrices that are symmetric. This poses only minor difficulties on virtual memory machines. Real memory machines (even with an extended memory) can place severe restrictions on problem size. Use of an extended memory (LCM on CDC) requires special modifications that go beyond the scope of this manual. Get a programmer's help! Further relaxation of these restrictions are possible but require significant program modifications.

An example of the size limitations (number of elements and nodes) for a representative real memory machine is given below:

Assume half bandwidth of 10 and $NE = NN$ with no rigid bodies. Also assume /IACOM/ is large enough and there are 13312 words of memory available, for /IACOM/. Then

$$\begin{aligned} NF3 &= 3*NN + 3*5 + 3*150 = 3*NN + 165 \\ NBASE &= 38*N + 9*NN + 21*(3*NN + 165) \\ &= (38 + 9 + 63) nn + 21*165 = 110*NN + 3465 \end{aligned}$$

for DEAD, LIVE, TSSS:

$$\begin{aligned} NF3 &= 3*NN \\ NEED &= NBASE + 10*NF3 \\ &= 110*NN + 3465 + 30*NN \\ &= 140*NN + 3465 = 13312 \end{aligned}$$

$$NN = \frac{13312 - 3465}{140} = 70$$

for DYN-DIM:

$$NEED = NBASE = 13312$$

$$NN = \frac{13312 - 3465}{110} = 89$$

for MODE:

$$NEED = 0.5 (3*3*NN*3*NN + 7*3*NN) = 0.5 (27 NN^2 + 21NN) = 13312$$

$$27*NN^2 + 21*NN - 2*13312 = 0$$

$$NN = \frac{-21 \pm \sqrt{21^2 - 8*27*13312}}{54} = 31$$

SEADYN uses the following files:

| FORTRAN | | |
|------------------|---------------|---|
| <u>File Name</u> | <u>Format</u> | <u>Use</u> |
| 01 | Binary | DEAD restart save file, "TAPE1" |
| 02 | Binary | LIVE, TSSS restart save file, "TAPE2" |
| 03 | Binary | DYN restart save file, "TAPE3" |
| 04 | Binary | Alternate restart input file, "TAPE4" |
| 05 | Formatted | System input file, "SEADYN.IN" |
| 06 | Formatted | System output file, "SEADYN.OUT" |
| 07 | Binary | Scratch file for temporary data storage |
| 08 | Formatted | Optional file for limit status reports, "LIMIT.DAT" |
| 09 | Binary | Scratch file for temporary data storage, "TAPE9" |
| 10 | Binary | Ship load data file, "SHIP.LOD" |
| 13 | Formatted | Scores RAO file, "SCORES.RAO" |
| or 13 | Formatted | External motion data file, "NODMOV.TAB" |
| 15 | Binary | Deciphered record images of free-form input, "TAPE15" |
| 16 | Formatted | Scratch file for rigid format input data, "TAPE16" |
| or 16 | Formatted | External plot file, "PLOT.DAT" |
| 17 | Binary | Hydrodynamic load coefficients for planned dynamic rigid bodies |
| 18 | Binary | Mode shape output, "MODE.DAT" |

(Files 11,12,14,18,19 are used by the proprietary interface to TDSIM)

DISTRIBUTION LIST

ARMY CECOM R&D Tech Lib, Ft Monmouth, NJ; R&D Lab, STRNC-UE, Natick, MA
ARMY CERL Library, Champaign, IL
CG FMF Pac, SCIAID (G5) Camp HM Smith, HI
COGARD R&DC Library, Groton, CT
COMDT COGARD Library, Washington, DC
COMOCEANSYS Lant, Code N9, Norfolk, VA
COMSC Washington, DC
COMSURFWARDEVGRU CO, Norfolk, VA
DOE Wind/Ocean Tech Div, Tobacco, MD
DTIC Alexandria, VA
DTRCEN Code 119, Annapolis, MD; Code 1250, Annapolis, MD; Code 1541 (Rispin), Bethesda, MD; Code 1561, Bethesda, MD; Code 1568, Annapolis, MD; Code 172, Bethesda, MD; Code 4111, Bethesda, MD; Code 4120, Annapolis, MD; Code 522, Annapolis, MD
LIBRARY OF CONGRESS Sci & Tech Div, Washington, DC
MARITIME ADMIN MAR-770 (Corkrey), Washington, DC; MMA, Library, Kings Point, NY
NAVCOASTSYSCEN CO, Panama City, FL; Code 2360, Panama City, FL; Code 715 (J. Mittleman) Panama City, FL; PWO (Code 740), Panama City, FL; Tech Library, Panama City, FL
NAVFACENGCOM Code 03, Alexandria, VA; Code 03T (Essoglou), Alexandria, VA; Code 04A, Alexandria, VA; Code 04A1, Alexandria, VA; Code 04A1D, Alexandria, VA; Code 04A3, Alexandria, VA; Code 04A3C, Alexandria, VA; Code 06, Alexandria, VA; Code 0631, Alexandria, VA; Code 07, Alexandria, VA; Code 07A (Herrmann), Alexandria, VA; Code 07M (Gross), Alexandria, VA; Code 09M124 (Lib), Alexandria, VA; Code 1002B, Alexandria, VA; Code 163, Alexandria, VA; Code 1651, Alexandria, VA
NAVFACENGCOM - CHES DIV. Code FPO-IC, Washington, DC; FPO-IPL, Washington, DC
NAVFACENGCOM - LANT DIV. Code 1112, Norfolk, VA; Code 405, Norfolk, VA; Library, Norfolk, VA
NAVFACENGCOM - NORTH DIV. Code 04AL, Philadelphia, PA
NAVFACENGCOM - PAC DIV. Library, Pearl Harbor, HI
NAVFACENGCOM - SOUTH DIV. Code 04A3, Charleston, SC; Code 102H, Charleston, SC; Code 406, Charleston, SC; Library, Charleston, SC
NAVFACENGCOM - WEST DIV. Code 04A2.2 (Lib), San Bruno, CA; Code 04B, San Bruno, CA
NAVOCEANO Code 6200 (M Paige), Bay St. Louis, MS; Library, Bay St Louis, MS
NAVOCEANSYSCEN Code 94, San Diego, CA; DET, R Yumori, Kailua, HI; DET, Tech Lib, Kailua, HI
NAVPGSCOL Code 1424, Library, Monterey, CA
NAVSEASYSCOM Code 56W23 (J Coon), Washington, DC; Code PMS296L22 (J Rekas), Washington, DC
NAVSWC DET, White Oak Lab, Code WSO, Silver Spring, MD
NOAA Data Buoy Off, Engrg Div, Bay St. Louis, MS; Joseph Vadus, Rockville, MD; Library, Rockville, MD
NORDA CO, Bay St. Louis, MS; Code 350, Bay St. Louis, MS; Code 352, Bay St. Louis, MS; Code 363, Bay St. Louis, MS; Code 410, Bay St. Louis, MS; Code 440, Bay St. Louis, MS
NRL Code 2511, Washington, DC; Code 4670 (B. Faraday), Washington, DC
NUSC DET Code 3322 (Brown), New London, CT; Code 2143 (Varley), New London, CT; Code 44 (RS Munn), New London, CT; Code TA131, New London, CT; Lib (Code 4533), Newport, RI
OCNR Code 1121 (EA Silva), Arlington, VA
US GEOLOGICAL SURVEY Gregory, Reston, VA; Marine Geology Offc (Piteleki), Reston, VA
USNA Ch, Mech Engrg Dept, Annapolis, MD; Ocean Engrg Dept (McCormick), Annapolis, MD
CATHOLIC UNIV of Am, CE Dept (Kim), Washington, DC
CLARKSON COLL OF TECH CE Dept (Batson), Potsdam, NY
CLEVELAND STATE UNIVERSITY CE Dept (Akinmusuru), Cleveland, OH
COLORADO SCHOOL OF MINES Dept of Engrg (Chung), Golden, CO
COLORADO STATE UNIVERSITY CE Dept (Criswell), Ft Collins, CO
CORNELL UNIVERSITY Civil & Environ Engrg (Dr. Kulhawy), Ithaca, NY; Library, Ithaca, NY
DUKE UNIVERSITY CE Dept (Muga), Durham, NC
FLORIDA ATLANTIC UNIVERSITY Ocean Engrg Dept (Su), Boca Raton, FL
FLORIDA INST OF TECH CE Dept (Kalajian), Melbourne, FL
GEORGIA INSTITUTE OF TECHNOLOGY CE Scol (Kahn), Atlanta, GA
INSTITUTE OF MARINE SCIENCES Library, Port Aransas, TX
IOWA STATE UNIVERSITY CE Dept (Handy), Ames, IA
JOHNS HOPKINS UNIV CE Dept (Jones), Baltimore, MD
LEHIGH UNIVERSITY CE Dept, Hydraulics Lab, Bethlehem, PA; Linderman Library, Bethlehem, PA; Marine Geotech Lab (A. Richards), Bethlehem, PA
MICHIGAN TECH UNIVERSITY CE Dept (Haas), Houghton, MI
MIT Engrg Lib, Cambridge, MA; Lib, Tech Reports, Cambridge, MA
NORTHWESTERN UNIV CE Dept (Dowding), Evanston, IL

PENNSYLVANIA STATE UNIVERSITY Applied Rsch Lab, State College, PA; Gotolski, University Park,
PA; Rsch Lab (Snyder), State College, PA
PORTLAND STATE UNIVERSITY Engrg Dept (Migliore), Portland, OR
PURDUE UNIVERSITY Engrg Lib, W. Lafayette, IN
STATE UNIVERSITY OF NEW YORK CE Dept (Reinhorn), Buffalo, NY; CE Dept, Buffalo, NY
TEXAS A&I UNIVERSITY Civil & Mech Engr Dept (Parate), Kingsville, TX
TEXAS A&M UNIVERSITY CE Dept (Herbich), College Station TX; Ocean Engr Proj, College Station, TX
UNIVERSITY OF CALIFORNIA CE Dept (Taylor), Davis, CA
UNIVERSITY OF DELAWARE CE Dept, Ocean Engrg (Dalrymple), Newark, DE
UNIVERSITY OF ILLINOIS Metz Ref Rm, Urbana, IL
UNIVERSITY OF MICHIGAN CE Dept (Richart), Ann Arbor, MI
UNIVERSITY OF NEW HAMPSHIRE Marine Prgm (Corell), Durham, NH
UNIVERSITY OF RHODE ISLAND CE Dept (Kovacs), Kingston, RI; CE Dept, Kingston, RI; Pell Marine
Sci Lib, Narragansett, RI
UNIVERSITY OF WISCONSIN Great Lakes Studies Cen, Milwaukee, WI
WOODS HOLE OCEANOGRAPHIC INST Doc Lib, Woods Hole, MA
AT&T TECH Eichler, Greensboro, NC
BATTELLE New Eng Marine Rsch Lab, Lib, Duxbury, MA
BECHTEL CIVIL, INC Woolston, San Francisco, CA
EARL & WRIGHT CONSULTING ENGRGS Jensen, San Francisco, CA
EASTPORT INTL, INC JH Osborn, Mgr, Ventura, CA
EG&G WASH ANAL. SVC CEN Bonde, Gaithersburg, MD
GOULD INC. Ches Instru Div, Tech Lib, Glen Burnie, MD
JOHN J MC MULLEN ASSOC Library, New York, NY
NATL ACADEMY OF ENGRG Alexandria, VA
PMB ENGRG Coull, San Francisco, CA
VSE Ocean Engrg Gp (Murton), Alexandria, VA
WESTERN INSTRUMENT CORP Ventura, CA
WESTINGHOUSE ELECTRIC CORP Library, Pittsburg, PA; Oceanic Div Lib, Annapolis, MD

NCEL DOCUMENT EVALUATION

You are number one with us; how do we rate with you?

We at NCEL want to provide you our customer the best possible reports but we need your help. Therefore, I ask you to please take the time from your busy schedule to fill out this questionnaire. Your response will assist us in providing the best reports possible for our users. I wish to thank you in advance for your assistance. I assure you that the information you provide will help us to be more responsive to your future needs.

R. N. Storer

R. N. STORER, Ph.D, P.E.
Technical Director

DOCUMENT NO. _____ TITLE OF DOCUMENT: _____

Date: _____ Respondent Organization : _____

Name: _____ Activity Code: _____
Phone: _____ Grade/Rank: _____

Category (please check):

Sponsor _____ User _____ Proponent _____ Other (Specify) _____

Please answer on your behalf only; not on your organization's. Please check (use an X) only the block that most closely describes your attitude or feeling toward that statement:

SA Strongly Agree A Agree O Neutral D Disagree SD Strongly Disagree

| | SA | A | N | D | SD | | SA | A | N | D | SD |
|--|-----|-----|-----|-----|-----|--|-----|-----|-----|-----|-----|
| 1. The technical quality of the report is comparable to most of my other sources of technical information. | () | () | () | () | () | 6. The conclusions and recommendations are clear and directly supported by the contents of the report. | () | () | () | () | () |
| 2. The report will make significant improvements in the cost and or performance of my operation. | () | () | () | () | () | 7. The graphics, tables, and photographs are well done. | () | () | () | () | () |
| 3. The report acknowledges related work accomplished by others. | () | () | () | () | () | | | | | | |
| 4. The report is well formatted. | () | () | () | () | () | | | | | | |
| 5. The report is clearly written. | () | () | () | () | () | | | | | | |

Do you wish to continue getting
NCEL reports?

☐ YES ☐ NO

Please add any comments (e.g., in what ways can we improve the quality of our reports?) on the back of this form.

Comments:

Please fold on line and staple

DEPARTMENT OF THE NAVY

Naval Civil Engineering Laboratory
Port Hueneme, CA 93043-5003

Official Business
Penalty for Private Use \$300



Code L03B
NAVAL CIVIL ENGINEERING LABORATORY
PORT HUENEME, CA 93043-5003

INSTRUCTIONS

The Naval Civil Engineering Laboratory has revised its primary distribution lists. The bottom of the label on the reverse side has several numbers listed. These numbers correspond to numbers assigned to the list of Subject Categories. Numbers on the label corresponding to those on the list indicate the subject category and type of documents you are presently receiving. If you are satisfied, throw this card away (or file it for later reference).

If you want to change what you are presently receiving:

- Delete - mark off number on bottom of label.
- Add - circle number on list.
- Remove my name from all your lists - check box on list.
- Change my address - line out incorrect line and write in correction (DO NOT REMOVE LABEL).
- Number of copies should be entered after the title of the subject categories you select.

Fold on line below and drop in the mail.

Note: Numbers on label but not listed on questionnaire are for NCEL use only, please ignore them.

Fold on line and staple.

DEPARTMENT OF THE NAVY

Naval Civil Engineering Laboratory
Port Hueneme, CA 93043-5003

Official Business
Penalty for Private Use, \$300

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 12503 WASH D.C.

POSTAGE WILL BE PAID BY ADDRESSEE

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

Commanding Officer
Code L34
Naval Civil Engineering Laboratory
Port Hueneme, California 93043-5003

DISTRIBUTION QUESTIONNAIRE

The Naval Civil Engineering Laboratory is revising its Primary distribution lists.

SUBJECT CATEGORIES

- 1 SHORE FACILITIES
- 2 Construction methods and materials (including corrosion control, coatings)
- 3 Waterfront structures (maintenance/deterioration control)
- 4 Utilities (including power conditioning)
- 5 Explosives safety
- 6 Aviation Engineering Test Facilities
- 7 Fire prevention and control
- 8 Antenna technology
- 9 Structural analysis and design (including numerical and computer techniques)
- 10 Protective construction (including hardened shelters, shock and vibration studies)
- 11 Soil/rock mechanics
- 14 Airfields and pavements
- 15 ADVANCED BASE AND AMPHIBIOUS FACILITIES
- 16 Base facilities (including shelters, power generation, water supplies)
- 17 Expedient roads/airfields/bridges
- 18 Amphibious operations (including breakwaters, wave forces)
- 19 Over-the-Beach operations (including containerization, material transfer, lighterage and cranes)
- 20 POL storage, transfer and distribution

TYPES OF DOCUMENTS

- 85 Techdata Sheets 86 Technical Reports and Technical Notes
83 Table of Contents & Index to TDS

- 28 ENERGY/POWER GENERATION
- 29 Thermal conservation (thermal engineering of buildings, HVAC systems, energy loss measurement, power generation)
- 30 Controls and electrical conservation (electrical systems, energy monitoring and control systems)
- 31 Fuel flexibility (liquid fuels, coal utilization, energy from solid waste)
- 32 Alternate energy source (geothermal power, photovoltaic power systems, solar systems, wind systems, energy storage systems)
- 33 Site data and systems integration (energy resource data, energy consumption data, integrating energy systems)
- 34 ENVIRONMENTAL PROTECTION
- 35 Hazardous waste minimization
- 36 Restoration of installations (hazardous waste)
- 37 Waste water management and sanitary engineering
- 38 Oil pollution removal and recovery
- 39 Air pollution
- 44 OCEAN ENGINEERING
- 45 Seafloor soils and foundations
- 46 Seafloor construction systems and operations (including diver and manipulator tools)
- 47 Undersea structures and materials
- 48 Anchors and moorings
- 49 Undersea power systems, electromechanical cables, and connectors
- 50 Pressure vessel facilities
- 51 Physical environment (including site surveying)
- 52 Ocean-based concrete structures
- 54 Undersea cable dynamics

- 82 NCEL Guides & Abstracts
91 Physical Security

☐ None-
remove my name